p7 ● 26

(7)

(19) JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **02098775 A**

(43) Date of publication of application: **11 . 04 . 90**

(51) Int. Cl **G06F 15/21**
G06F 15/18
G06F 15/31

(21) Application number: **63251334**

(22) Date of filing: **04 . 10 . 88**

(71) Applicant: **NEC CORP**

(72) Inventor: **YAMANISHI KENJI**

(54) **METHOD AND DEVICE FOR GENERATING DECISION LIST**

(57) Abstract:

PURPOSE: To perform the classification and prediction of data outputted frequently from an information source with high accuracy by picking up the data from decision which maximizes an information gain, and classifying the data preferentially from the decision with high identification capacity.

CONSTITUTION: The decision with high information gain for measuring data is added sequentially as the decision of a decision list as a logical product which regulates the decision of the decision list under a state where the number of characters comprising one logical product is fixed. And a stopping rule based on an error classification rate is provided, and the addition of the decision is stopped at a point where it is satisfied, then, the decision list obtained at that time is outputted. In such a way, it is possible to classify and predict unknown data generated from the same information source as that for the measuring data with high accuracy.

(19)日本国特許庁（ＪＰ）　(12)特　許　公　報（Ｂ２）　(11)特許番号

第2581196号

(45)発行日　平成９年(1997)２月12日　(24)登録日　平成８年(1996)11月21日

(51)Int.Cl.[6]	識別記号	庁内整理番号	ＦＩ		技術表示箇所
Ｇ０６Ｆ　17/60			Ｇ０６Ｆ　15/21	Ｚ	

請求項の数５(全 16 頁)

(21)出願番号　特願昭63－251334

(22)出願日　昭和63年(1988)10月４日

(65)公開番号　特開平2－98775

(43)公開日　平成２年(1990)４月11日

(73)特許権者　999999999
日本電気株式会社
東京都港区芝５丁目７番１号

(72)発明者　山西　健司
東京都港区芝５丁目33番１号　日本電気株式会社内

(74)代理人　弁理士　京本　直樹　(外２名)

審査官　石井　茂和

(54)【発明の名称】　決定リストの生成方法及び装置

(57)【特許請求の範囲】

【請求項１】複数の属性とクラスの組として与えられる複数の観測データから、決定項の順序付きリストである決定リストを生成させる方法において、

決定項に対応する論理積の論理変数の数を固定したもとで、あらゆる論理積の中で、観測データに対する情報利得を計算して、これが最大になるような論理積を選択し、リストの末尾に該論理積に対応する決定項を付加することを順次繰り返すステップと、

予め定めたストッピングルールによって決定項の付加を終了したところで得られる決定リストの末端から決定項を１つづつ取り除くことによって得られる決定リストの系列の１つ１つに対して、決定リストによって分類された観測データを２元符号化するのに必要な符号長と決定リスト自身を２元符号化するのに必要な符号長の両者を

合わせた総符号長として定義される記述量を計算し、該決定リスト系列の中で記述量を最小にする決定リストを最終的に求める決定リストとして出力するステップと、

を含むことを特徴とする決定リストの生成方法。

【請求項２】請求項１に記載の決定リストの生成方法によって各固定されたｋに対して決定リストを生成するステップと、

様々なｋに対して生成された決定リストの中から、決定リストによって分類された観測データを２元符号化するのに必要な符号長と、決定リスト自身を２元符号化するのに必要な符号長の両者を合わせた総符号長として定義される記述量を計算し、最小の記述量を有する決定リストを最終的に求める決定リストとして出力するステップと、

を含むことを特徴とする決定リストの生成方法。

【請求項３】複数の属性とクラスの組として与えられる複数の観測データから、決定項の順序付きリストである決定リストを生成させる装置において、
観測データを記憶する手段と、
決定項に対応する論理積の論理変数の数を固定したもとで、あらゆる論理積の中で、観測データに対する情報利得を計算して、これが最大になるような論理積を選択し、リストの末尾に該論理積に対応する決定項を付加することを順次繰り返す手段と、
予め定めたストッピングルールによって決定項の付加を終了したところで得られる決定リストを最終的に求める決定リストとして出力する手段と、を含むことを特徴とする決定リストの生成装置。

【請求項４】前記請求項３に記載の決定リストの生成装置により一度決定リストを生成する手段と、
生成された決定リストの末端から決定項を１つづつ取り除くことによって得られる決定リストの系列の１つ１つに対して、決定リストによって分類された観測データを２元符号化するのに必要な符号長と決定リスト自身を２元符号化するのに必要な符号長の両者を合わせた総符号長として定義される記述量を計算し、該決定リスト系列の中で記述量を最小にする決定リストを最終的に求める決定リストとして出力する手段と、を含むことを特徴とする決定リストの生成装置。

【請求項５】前記請求項３又は４に記載の決定リストの生成装置により各固定されたｋに対して決定リストを生成する手段と、
様々なｋに対して生成された決定リストの中から、決定リストによって分類された観測データを２元符号化するのに必要な符号長と決定リスト自身を２元符号化するのに必要な符号長の両者を合わせた総符号長として定義される記述量を計算し、最小の記述量を有する決定リストを最終的に求める決定リストとして出力する手段と、を含むことを特徴とする決定リストの生成装置。

【発明の詳細な説明】

（産業上の利用分野）

この発明は複数の属性とクラスの組として与えられる、雑音を伴う複数と観測データから決定リストを発生させる方法に関する。

（従来の技術）

複数の属性とクラスの組として与えられる観測データから属性とクラスの構造的な関係を生成し、それを表現するための方法として決定リストによる分類規則生成の方法がある。決定リストの概念については、1987年発行の米国の雑誌「マシンラーンニング（Machine Learning）」の２巻の中の229－246頁掲載のR.Lリベスト（R.L. Rivest）による論文「ラーニングデシジョンリスト（Learnig decision lists）」に記載されており、２値データを扱う限りにおいては現在知られている分類規則の表現能力の高いものであることがわかっている。この論文の中では、雑音の伴わない観測データから、全ての観測データにつじつまを合わせるような決定リストを構成する方法が記載されている。

（発明が解決しようとする課題）

前記論文で示された決定リストの生成方法は、雑音の伴わない観測データから全ての観測データにつじつまを合わせるような決定リストを構成するための方法であった。しかし、実際に扱い観測データは一般に雑音や曖昧性などの不確実さを伴うので、全ての観測データを説明出来なくても、同じ情報源から発生するデータを出来るだけ正しく分類するような決定リストの方法が必要であるのだか、このような決定リストを発生させる手段は存在していなかった。

本発明の目的は雑音を伴う観測データから、未知データに対する予測誤差が最小になるような決定リストを自動的に発生させる方法を提供することにある。

（課題を解決するための手段）

本発明による決定リストの生成方法の１つは、決定項に対応する論理積の論理変数の数を固定したもとで、あらゆる論理積の中で、観測データに対する情報利得を計算して、これが最大になるような論理積を選択し、リストの末尾に該論理積に対応する決定項を付加することを順次繰り返すステップと、予め定めたストッピングルールによって決定項の付加を終了したところで得られる決定リストを最終的に求める決定リストとして出力するステップと、を含むことを特徴とする、方法である。

あるいは、上記発明に対して次のような変形を施すことも出来る。上の方法によって一度決定リストを生成するステップと、次に該決定リストの末端から決定項を１つづつ取り除くことによって得られる決定リストの系列の１つ１つに対して、決定リストによって分類された観測データを２元符号化するのに必要な符号長と決定リスト自身を２元符号化するのに必要な符号長を両者を合わせた総符号長として定義される記述量を計算し、該決定リスト系列の中で記述量を最小にする決定リストを最終的に求める決定リストとして出力するステップと、を含むことを特徴とする方法である。（上の２つの方法を「発明１」とする。）

また、本発明によるもう１つの決定リストの生成方法は、発明１の方法によって各固定されたｋに対して決定リストを生成するステップと、様々なｋに対して発明１の方法によって生成された決定リストの中から、決定リストによって分類された観測データを２元符号化するのに必要な符号長と決定リスト自身を２元符号化するのに必要な符号長の両者を合わせた総符号長として定義される記述量を計算し、最小の記述量を有する決定リストを最終的に求める決定リストとして出力するステップと、を含むことを特徴とする決定リストの生成方法である。（以下、この方法を「発明２」とする。）

本発明による決定リストの生成装置の１つは、観測デ

5

ータを記憶する手段と、決定項に対応する論理積の論理変数の数を固定したもとで、あらゆる論理積の中で、観測データに対する情報利得を計算して、これが最大になるような論理積を選択し、リストの末尾に該論理積に対応する決定項を付加することを順次繰り返す手段と、予め定めたストッピングルールによって決定項の付加を終了したところで得られる決定リストを最終的に求める決定リストとして出力する手段と、を含むことを特徴とする、装置である。

また、本発明の決定リスト生成装置に対して次のような変形を施すことも出来る。上記装置によって一度決定リストを生成する手段と、次に該決定リストの末端から決定項を１つづつ取り除くことによって得られる決定リストの系列の１つ１つに対して、決定リストによって分類された観測データを２元符号化するのに必要な符号長と決定リスト自身を２元符号化するのに必要な符号長の両者を合わせた総符号長として定義される記述量を計算し、該決定リスト系列の中で記述量を最小にする決定リストを最終的に求める決定リストとして出力する手段と、を含むことを特徴とする決定リストの生成装置である。（上の２つの装置を「発明３」とする。）

また、本発明によるもう１つの決定リスト生成装置は、発明３の装置によって各固定されたｋに対して決定リストを生成する手段と、様々なｋに対して発明３の装置によって生成された決定リストの中から、決定リストによって分類された観測データを２元符号化するのに必要な符号長と決定リスト自身を２元符号化するのに必要

$$(x_1\overline{x}_2, 1),\quad (x_2\overline{x}_3\overline{x}_5, 0),\quad (x_3x_4, 1),\quad (\mathrm{true}, 0) \qquad (2)$$

ここで、（２）は（１）で定義した決定リストにおいて$r=3$、$t_1=x_1\overline{x}_2$ $t_2=x_2\overline{x}_3\overline{x}_5$、$t_4=$trueを代入したものである。

この決定リストによるデータと分類の決定過程は第７図のように表される。

第７図で示したように、決定リストは“if－then－else if－then－”型の概念分類規則の一般化となっている。ｎ型の属性と１つのクラスで記述された複数の観測データから同じ情報源より発生するデータを出来るだけ正確に予測するような決定リストを構成するには、どのような方法あるいは装置によって決定リストを発生させたらよいかということが問題であり、この問題に答えるのが本発明である。但し、観測データには雑音や曖昧さなどの不確実性が含まれているものとする。先ず、この問題に答える１つの方法として、観測データに対する情報利得（エントロピー、誤分類率、Gini指標等の減少分で図られる）の高い決定から順に決定リストの決定として加えていくようにし、誤分類率に基づくストッピングルールを設けて、これが満足されたところで決定の追加を停止し、そこで得られる決定リストを出力するという方法、あるいはさらにそこで得られる決定リストの末端から決定項を順次取り除くことによって得られる決定リ

6

な符号長の両者を合わせた総符号長として定義される記述量を計算し、最小の記述量を有する決定リストを最終的に求める決定リストとして出力する手段と、を含むことを特徴とする決定リストの生成装置である。（この装置を「発明４」とする。）

（作用）

先ず、決定リストについて説明する。以下では簡単のため変数の値は全て｛0,1｝であるとするが、一般には変数値は離散多値でも連続値であってもよい。今、｛0,1｝に値をとる変数の数をｎとし、ｋはｎ以下の正の整数であるとする。$L_n=\{x_1, \overline{x}_1, \cdots, x_n, \overline{x}_n\}$（$\overline{x}_i = 1-x_i$（$i=1,\cdots,n$））とし、$L_n$の元をリテラルとよび、リテラルの論理積をタームとよぶ。ｋ－DL（ｎ）はT^n_kの元t_j（$j=1,\cdots,r$）と｛0,1｝に値をとるv_j（$j=1,\cdots,r$）の組のリストとして以下のように表されるものの集合を表す。ここに、T^n_kはL_nのうち高々異なるｋ個のリテラルで表されるターム全体の集合を示し（但し、x_iと同時に$\overline{x}_i$を含まない）、最後の関数t_rはtrueを返す定数関数である。

$$(t_1, v_1),\ \cdots,\ (t_r, v_r) \qquad (1)$$

１つの決定リストは任意の$x \in X_n$（X_nはｎ桁のブーリアンベクトルの全体の集合を表す）に対して$t_j(x)=1$となる最初のｊに対するv_jの値を示す。(t_i, v_i)を第ｉ決定（あるいは単に決定）と呼ぶ。例えば、次は３－DL（４）（ｋ－DL（ｎ）においてｋ＝３、ｎ＝４を代入したもの）の元である。

ストの系列の要素の１つ１つに対して、その決定リストを用いて記述される観測データの記述量（観測データを決定リストによって予め分類してから、決定リスト自信も含めて２元符号語に符号化した際の符号長）を計算し、該決定リスト系列の中で記述量を最小にする決定リストを最終的に求める決定リストとする方法が考えられる。情報利得を最大にする決定から拾っていく理由は、識別能力が高い決定から先にデータを分類することにより、情報源から頻出するデータについて、より精度の高い分類・予測が行なえるからである。また、記述量を最小にする決定リストを選ぶことの理由は、観測データをモデルを用いて自己完結的な符号化を行う際に、全データをより圧縮しうるモデル（この場合は決定リスト）は、同じ情報源より発生する未知の観測データに対し、より正確な予測を行えるモデルであるということが漸近的に成立するからである。この理論的裏付けに関しては、特に統計モデルに適用する場合に関しては1986年発行の米国の雑誌アナルス　オブ　スタティスティクス（Annals of Statistics）の14巻の1080－1100頁掲載のJ.リサネン（J.Rissanen）による論文「ストキャスティック　コンプレキシティー　アンド　モデリング（Stochastic complexty and modeling）」にMDL基準として述

7

べられている。しかし、本発明では、分類規則としての決定リストといった論理型概念学習の最適モデル化にMDL基準を用いているところに新規性をおいている。以上が発明１の方法の原理であり、おなじ原理を装置の形で

8

実現するのが発明３である。上述の情報利得を以下に正確に定義する。先ず、以下のような記号を定める。

［記号］　以下、対数の底は全て２とする。

・k, n∈N（自然数全体）は固定とする。

・ $S\triangleq\{<i,x,c>$(学習データ): i(対象番号)$\in N$,

x（属性値）∈ $\{0,1\}^n$, c（クラス）∈ $\{0,1\}$ }：学習データの集合（属性値は属性変数$x_1,\cdots,x_n$に対するデータの値を示す。）

・SA（t）＝ $\{<i,x,c>\in S: t(x)=1\}$, $t\in T^n_k$

SB（t）＝ $\{<i,x,c>\in S: t(x)=0\}$, $t\in T^n_k$

・A（t）＝#SA（t）（以下、#Wは集合Wの元の総数を表す）

B（t）＝#SB（t）

・$A^+(t)=\#\{<i,x,c>\in SA(t): c=1\}$, $t\in T^n_k$

$A^-(t)=\#\{<i,x,c>\in SA(t): c=0\}$, $t\in T^n_k$

・$B^+(t)=\#\{<i,x,c>\in SB(t): c=1\}$, $t\in T^n_k$

$B^-(t)=\#\{<i,x,c>\in SB(t): c=0\}$, $t\in T^n_k$

ここに、A（t）＝$A^+(t)+A^-(t)$, B（t）＝$B^+(t)+B^-(t)$とする。

$$I(t)\triangleq\frac{A(t)}{A(t)+B(t)}I_A(t)+\frac{B(t)}{A(t)+B(t)}I_B(t) \quad (3)$$

$$\cdot\ \Delta I(t)\triangleq I_B(t'')-I(t) \quad (4)$$

を決定（t, v）による情報利得とよぶ。

（t″は決定リストにおいてtの直前に現れる決定のタームとする。）

ここで、$I_A(t)$, $I_B(t)$の選び方としては次のようなものを考えることが出来る。

（i）エントロピー

$$I_A(t)\triangleq-\frac{A^+(t)}{A(t)}\log\frac{A^+(t)}{A(t)}-\frac{A^-(t)}{A(t)}\log\frac{A^-(t)}{A(t)}$$

$$I_B(t)\triangleq-\frac{B^+(t)}{B(t)}\log\frac{B^+(t)}{B(t)}-\frac{B^-(t)}{B(t)}\log\frac{B^-(t)}{B(t)} \quad (5)$$

(ii) Gini指標

$$\cdot I_A(t)\triangleq\frac{A^+(t)}{A(t)}\cdot\frac{A^-(t)}{A(t)}$$

$$\cdot I_B(t)\triangleq\frac{B^+(t)}{B(t)}\cdot\frac{B^-(t)}{B(t)} \quad (6)$$

(iii) 誤分類率

$$\cdot I_A(t)\triangleq\frac{\min\{A^+(t),A^-(t)\}}{A(t)}$$

$$\cdot I_B(t)\triangleq\frac{\min\{B^+(t),B^-(t)\}}{B(t)} \quad (7)$$

情報利得ΔI（t）の計算方法としては、上の他に、

ΔI（t）＝－H｛A^+（t）/A（t）＋δ｝・1/log（A（t）＋1）

9　　10

や

$$\Delta I(t) = -\{H((A^+(t)+1)/(A(t)+2))+\delta\}\cdot 1/\log(A(t)+1)$$

や

$$\Delta I(t) = A(t) - A(t)\cdot H(A^+(t)/A(t))$$

や

$$\Delta I(t) = \{A(t) - A(t)\cdot H(A^+(t)/A(t))\}\cdot 1/\log(A(t)+1)$$

などいろいろなものがとりうるものとする。但し、上に於て、δはある定数、H（x）＝－xlogx－（1－x）log（1－x）：エントロピー関数,A（t）＝A^+（t）＋A^-（t）とする。

次に、決定リストを用いた全観測データの記述量計算の例をあげる。データの記述量は、決定リスト自体の記述量と決定リストによって分類されるデータの中の例外の記述量との和あるいは何らかの補正を施して結合した量として与えられる。例えば、（2）の決定リストについては、T^5_3＝131（集合T^5_3の全要素の数が131であることを意味する）である（定数関数trueに対応するタームを0としてこれも数える）から、（2）の最初の決定のタームは1/131の確率で選ばれているので、これを自己完結的に符号化するためには、log（131）bitsの符号長が必要である。この場合、符号化方法としては、Huffman符号化を用いる。また、1つの決定に対しては、タームを真にするようなデータに1を割り付けるか0を割り付けるかも記述しなければならず、これには1bit必要であるから、結局（2）の最初の決定に関する記述量として、log（131）＋1bits必要である。次の決定に関しては、T^5_3の最初の決定に用いられたタームを除く130個のタームの中から決定に必要なタームが選ばれるのであるから、同様にして、log（130）＋1bitsの記述量が必要である。同様にして、各決定に対する記述量を求め、それらの総和を計算することにより決定リスト自体の記述量が計算できる。（2）に対しては全部で（log（131）＋1）＋（log（130）＋1）＋（log（129）＋1）＋（log（128）＋1）bits必要である。また、例外データの記述量は、例えば、1つの決定に対して、その決定値が1であるとして、その決定におけるタームを真にするデータが7つであり、そのうちクラスが1であるものの数が5,0であるものの数が2であるとして、対象番号の若い順にデータのクラスを記述すると、1101101であったとする。このとき、この系列を自己完結的に符号化するのに必要な符号長は、log（7＋1）＋log（$_7C_2$）bitsまたは初めから例外は必ず

$$\lfloor (7+1)/2 \rfloor$$

個以下であることが分かっていれば、

$$\log(\lfloor (7+1)/2 \rfloor + 1) + \log({}_7C_2)\text{bits}$$

である。一般に、系列の長さをN, b＝b_1またはb_2、ここにb_1＝N,

$$b_2 = \lfloor (N+1)/2 \rfloor$$

とし、例外の数をhとすることにより、例外記述に必要な記述量は

log（b＋1）＋log（$_NC_h$）bits　　（8）

または、

L_N（h）＋log（$_NC_h$）bits　　（9）

で与えられる。ここに、L_N（h）は{0,1,…,N}に含まれる自然数を一意的に復号できるように符号化を行うときの符号長を表し、次に満たす。

$L_N(0) = 1$

$L_N(k) = 1 + \log k + \log\log k + \cdots C_N \quad (k > 1)$　　（10）

但し、上の和は正の項のみに対してとられるものであり、C_Nは

$$\sum_{k=0}^{M} 2^{-L_N(k)} = 1$$

を満たす実数である。

記述量の計算は符号化の仕方に依存するので、上述の計算方法はあくまで例であり、これ以外にも、考えらえる一意復号可能な符号化に対応して様々な記述長計算方法が用いられて良いものとする。

以上に与えた方法によって算出される決定リスト自身の記述量と例外データの記述量との和が決定リストを用いることによる全学習データの記述量である。あるいは決定リストの記述量と例外データの記述量のいずれかに補正を施した量を結合した利用を記述量として用いる場合もある。上で述べた情報利得と記述量の概念を用いて、所与の観測データに対する決定リストの最適化を行うことが出来る。例えば、属性数が6の第9図に示すような学習データを考える。これらの中に、たとえば、9と25のように属性値が同じでクラスが異なるといったデータの衝突が複数見られる。発明1によれば、先ず、kの値を固定したときの決定リストを、情報利得を最大にするタームtに対して、A^+（t）≧A^-（t）ならば（t,1）を、そうでないならば（t,0）を逐次的に付加して行く方法で構成し、予め定めたストッピングルールで停止して、そこで得られる決定リストを求める決定リストとして出力することにより、次のような決定リストが得られる（k＝2,3に対し、それぞれDL*（2）,DL*（3）とかく）。但し、情報利得はエントロピーを用いるもの

とし、このときの決定付加の停止条件としてたとえば、前決定における決定値をｖ、情報利得を最大にするタームをｔとして、次のｉ），ii），iii）の条件と採用するものとする。

ｉ）　$\min\{A^+(t), A^-(t)\}/A(t) > \alpha$ （＝0.25）ならば、

$B^+(t) \geqq B^-(t)$ ならば（true, 1）を、そうでないならば（true, 0）を最終決定として付加して停止する。

ii）　$\min\{B^+(t), B^-(t)\}/B(t) \leqq \beta$ （＝0.29）ならば、

$B^+(t) \geqq B^-(t)$ ならば（true, 1）を、そうでないならば（true, 0）を最終決定として付加して停止する。

iii）　まだ決定されていない観測データがもうなければ、

（true, 1－ｖ）を決定として右に加えて出力し、停止する。

DL＊（２）は次のとおり、

$(x_3x_5, 0)$　$(x_2x_5, 0)$　$(x_1x_4, 1)$　(true, 1)

DL＊（３）は次のとおり、

$(x_3x_5, 0)$　$(x_2x_5, 0)$　$(x_1x_5x_6, 1)$　(true, 1)

なお、上述のストッピングルールの決め方は、この方法にのみ限定はされない。

また、発明３によれば、観測データを記憶する手段と、情報利得を最大にする決定から順に付加して、予め定めているストッピグルールの条件が満たされたときに得られる決定リストを求める最終的な決定リストとして出力として出力する手段を具備している装置によって、ｋ＝2,3のときにはそれぞれDL＊（２），DL＊（３）が発生させられる。

尚、以上の決定リストの発生方法並びに装置では、固定されたｋに対してのみｋ－DL（ｎ）が発生させることが出来る。一般にｋの値が大きければ大きいほど表現能力が高くなり、多次元のデータ空間の分割の制度も細かくなるが、観測データには一般に雑音や曖昧さなどの不確実性をともなうが入っているので、ｋが必要以上に大きいと、統計的な揺らぎに過敏な決定リストを構成してしまうことになり、その場合、未知データに対してする分類予測誤差は返って大きくなる。従って、同じ情報源から発生する未知データに対する予測誤差を最小にするような最適なｋの値が存在するはずであり、このようなｋに対する決定リストを発生させる方法並びに装置が必要となる。上述の意味で最適な決定リストを発生させるためには、様々なｋの値に対して、発明１の方法または発明は３の方法によって決定リストを発生させてから、それらを用いて記述される観測データの記述量を計算して比較し、記述量を最小にするような決定リストを最終的に求める決定リストとして求める方法並びに装置が考えられる。この理論的根拠も前出のJ.リサネンによる論文に示されているMDL基準の考え方に依るものである。以上が発明２の方法の原理であり、同じ原理を装置の形

で実現するのが発明４である。例えば、前出の例では、$\#T^6{}_2 = 73$, $\#T^6{}_3 = 233$であり、各決定における（決定される対象の数、例外の数）は、

DL＊（２）で、

(6, 0)，(3, 0)，(7, 0)，(19, 6)

DL＊（３）で、

(6, 0)，(3, 0)，(2, 0)，(24, 4)

であるから、記述量は、例外の記述量の計算方法としてはｂ＝b_1として（８）を用いることにすれば、次のように求められる。

DL＊（２）で、

決定リストの記述量＝28.639bits

例外の記述量＝26.783bits

総記述量＝55.422bits

DL＊（３）で、

決定リストの記述量＝35.419bits

例外の記述量＝24.411bits

総記述量＝59.831bits

従って、MDL基準の下ではDL＊（２）がDL＊（３）よりも適当なモデルであると言うことが出来る。発明２によると、ｋ＝２とｋ＝３に対して発明１の方法で１度決定リストを発生させ、さらに各ｋに対する記述量を計算し、それらを比較して小さい記述量を与える決定リストとして、DL＊（２）を発生させることが出来る。発明４によると、ｋ＝２とｋ＝３に対して発明３の装置で１度決定リストを発生させる手段と、各ｋに対する記述量を計算する手段と、それら及び各ｋに対する決定リストを記憶する手段と、記述量を比較し、最小記述量をもつ決定リストを出力する手段を具備していれば、DL＊（２）を発生させることが出来る。

（実施例）

次に、本発明について図面を参照して詳細に説明する。以下、記号は（作用）の項に従う。

第１図は発明１の実施例を説明するフローチャートである。startでは、対象番号と２値のｎ個の属性と２値のクラスからなる観測データの前集合を初期値とする集合Ｓと、固定されたn, kに対して$T^n{}_k$を初期値とする集合Ｔと、初期タームｔ″＝０（ターム０は$T^n{}_k$の中に含まれないが、全データを偽にするタームとして定める。また、このタームを用いた決定は出力時点では省略されるものとする）と、初期決定値ｖ＝０が与えられている。ステップ11で、ストッピングルールの条件として、最初から与えられている１以下の正の実数α，βに対して、

ｉ）　Ｓ＝空集合

ii）　$\min\{A^+(t''), A^-(t'')\}/A(t'') > \alpha$

iii）　$\min\{B^+(t''), B^-(t'')\}/B(t'') \leqq \beta$

を設けて、ｉ）ii）iii）の順にこれらの条件の１つでも当てはまるかどうかを調べる。これらの条件のうち一つでもあてはまるものがあれば、ステップ12に進み、最終決定として$B^+(t) \geqq B^-(t)$ ならば（true, 1）を、

13

そうでないならば（true,0）を付加して、それまで付加してきた決定と併せて出力して終了する。これらの条件がいずれも満たされていないならば、ステップ13に進み、（4）で定められている情報利得をタームtの関数とみなしたときにこれを最大にするタームtをTの中から決定する。（これをt*とする。）情報利得最大のタームが複数ある場合は、その中でランダムにt*を選ぶ。次に、ステップ14で、$A^+(t^*)>A^-(t^*)$ならばv＝1とし、$A^+(t^*)<A^-(t^*)$ならばv＝0とし、(t^*,v)を決定リストの決定として右に付け加える。次に、ステップ15で、タームtを真にするデータをSから除いた集合を改めてSとし、タームtをTから除いた集合を改めてTとし、t*を改めてt″とし、ステップ11に戻る。

第２図は発明１のもう１つの実施例を説明するフローチャートである。ステップ21から25まではそれぞれ、第１図のステップ11から15までと同じ機能を果たし、ステップ25の出力として、第１図の出力と同じ決定リストが得られる。第２図では、ステップ25の出力としての決定

14

リスト（$DL_{max}(k)$とする）に対して、さらにステップ26で、$DL_{max}(k)$を用いて記述できる観測データの記述量を（作用）の項で述べたような方法で決定リスト自身の記述量と決定に対する例外データの記述量との和として計算し、これをL_0とする。ここで、記述量は、決定リスト自身の記述量と決定に対する例外データの記述量との単純な和ではなく、重み付き和、すなわち、λ（決定リストの記述量）＋（１－λ）（例外データの記述量）（０＜λ＜１）として計算をする場合あるいは決定リストの記述量と例外データの記述量のいずれかを補正して結合した量を用いる場合もあるとする。以下の記述量計算においても同様である。次に、ステップ27で、$DL_{max}(k)$の末端（最終）決定から刈り込みを行う。具体的には、V（１）＝$DL_{max}(k)$として、j≧１に対し、V（j）の右から２つの決定を（t,v）（true,v′）とするとき、これらを（true,v*）に置き換えたものをV（j＋１）とする。但し、t′をtの直前の決定に用いられたタームとすると、v*は次のように与えられる。

$$V^*=\begin{cases}1 & (A^+(t)+B^+(t)\geqq A^-(t)+B^-(t)\text{の場合})\\ 0 & (\text{それ以外の場合})\end{cases}$$

次に、ステップ28で、V（j＋１）を用いて記述される観測データの記述量を（作用）の項に示した方法で計算し、これをL_{j+1}とする。次に、ステップ29に進み、V（j＋１）からさらに切り取る決定が残っているかどうかを判断し、残っていなければ、ステップ31に進み、記述量全てのL_jの値を比較して最小になるものに対するV（j）を出力し、終了する。ここで、もし、最小値が複数存在すれば、その最小値に対応するV（j）を複数出力し、終了する。また、ステップ29においてV（j＋１）から刈り取る決定が残っていれば、ステップ30に進み、jをj＋１としてステップ27に戻る。

第３図は発明２の実施例を説明するフローチャートである。startにおいては、対象番号と２値のｎ個の属性と２値のクラスとからなる観測データの全集合を初期値とする集合Sと、固定されたｎ、と複数のｋに対して（ｋは１つの決定に用いられる論理積を構成する文字の最大数であり、ｋのとりうる数の総数をMとする）、T^n_kを初期値とする集合T（ｋ）と、初期タームt″＝０と、初期値決定値v＝０が与えられている。

先ず、各ｋに対して、発明１の方法で決定リストを発生させる。$k_1<\cdots<k_M$を対象にするｋとして、k_jに対応する決定リスト発生のステップを31とする。ステップ31は第１図の全体或は第２図の全体である。次に、発生した決定リストに対して、その決定リストを用いて記述されるデータの記述量を計算する。第２図で示された方法をステップ31にする場合はこのステップは省略される。k_iに対応する記述量計算のステップをステップ32で表す。次に、ステップ33各ｋの値に対して計算された記述量を比較して、最小値を求める。ここで、記述量は、決定リストの記述量と決定に対する例外データの記述量との単純な和ではなく、重み付き和、すなわち、λ（決定リストの記述量）＋（１－λ）（例外データの記述量）（０＜λ＜１）として計算する場合、あるいは決定リストの記述量と例外データの記述量のいずれかを補正して結合した量を用いる場合もあるとする。次に、ステップ34で、これらの中で最小なｋの値に対応する決定リストを出力し、終了する。第４図は発明３の装置を示すブロック図である。対象番号と２値のｎ個の属性と２個のクラスとからなる観測データの全集合を入力として、これを41の記憶装置で一旦記憶する。制御信号発生装置から発生する制御信号の指令によって記憶装置から観測データの一部が情報利得最大ターム決定回路42に送られる。最初に供給されるデータは観測データ全部である。情報利得最大ターム決定回路42は初期状態のパラメータとして、$T=T^n_k$（ｋは固定）、t″＝０を有しており、

i)　S＝空集合

ii)　$\min\{A^+(t''),A^-(t'')\}/A(t'')>\alpha$

iii)　$\min\{B^+(t''),B^-(t'')\}/B(t'')\leq\beta$

の条件をi) ii) iii)の順に調べ、１つでも当てはまるような場合は、信号P及びt″を42に従属する決定付

加回路43に送る。また、上の条件がいずれも満たされない場合には、固定されたｋの値に対して、Ｔの中のタームについて、記憶装置から供給されるデータに対して（４）で与えられる情報利得を計算し、これを最大にするようなタームｔ＊を決定し、状態をｔ″＝ｔ＊，Ｔ＝Ｔ－ｔ＊に変えて、ｔ＊を決定付加回路43に送る。決定付加回路43は、情報利得最大ターム決定回路42の出力と記憶装置から供給されるデータ（42に供給されるデータと同じ）を入力とし、42からｔ＊が送られてきた場合はA^+（ｔ＊），A^-（ｔ＊）を計算して、A^+（ｔ＊）＞A^-（ｔ＊）ならば、ｖ＝1，A^+（ｔ＊）＜A^-（ｔ＊）ならば、ｖ＝０とし、（ｔ＊，ｖ）を決定リストの決定として右に付け加え、次に記憶装置41がＳ－SA（ｔ＊）を42に供給するように指令する制御信号を発生させる信号を、制御信号発生装置44に送る。44は記憶装置がＳ－SA（ｔ＊）を42に供給するように指令する制御信号を発生させる信号を41に送る。44の指令を受けて新たに記憶装置41からデータが供給されると、42は、同じ動作を繰り返す。43に42から信号Ｐが送られてきた場合には、B^+（ｔ″）≧B^-（ｔ″）ならば、ｖ＝１、そうでないならば、ｖ＝０として、（true，ｖ）を最後の決定として決定リストの右に付け加えて、回路全体の出力としてこれまで決定を付加して得られた決定リストを出力し、回路全体の動作を終了する。

第５図は発明３の決定リストの発生装置のもう１つの実施例を示すブロック図である。回路51，52，53，54はそれぞれ、第４図の回路41，42，43，44と同じ入出力機能を果たする。しかし、回路53は、第４図の43の出力と同じ決定リストにくわえて、各決定に用いられたタームのそれぞれに対するA^+（ｔ），A^-（ｔ）の値も出力する。刈り込み及び記述量計算回路55は、回路53の出力を入力として、それを末端の決定から順に刈り込んで、それらを用いてデータを記述するときの記述量を併せて計算する。ここで、記述量は、決定リスト自身の記述量と決定に対する例外データの記述量との単純な和ではなく、重み付き和、すなわち、λ（決定リストの記述量）＋（１－λ）（例外データの記述量）（０＜λ＜１）として計算する場合、あるいは決定リストの記述量と例外データの記述量のいずれかを補正して結合した値を用いる場合もあるとする。具体的には、右から２つの決定を（ｔ，ｖ）（true，ｖ′）とするとき、これらを（true，ｖ＊）に置き換えるといった操作を刈り取る決定が無くなるまで繰り返す。但し、ｖ＊は次のように与えられる。

$$V^*=\begin{cases}1 & (A^+(t')+B^+(t')\geqq A^-(t')+B^-(t')\text{の場合})\\ 0 & (\text{そうでない場合})\end{cases}$$

55はそれぞれ刈り込みによって得られた決定リストのそれぞれと、それぞれに対して計算された記述量を併せて、第２記憶装置56に送る。第２記憶装置は55から送られてきた入力を記憶する。また、55は刈り込まれたそれぞれの決定リストの記述量だけを記述量比較回路57にも送る。57は刈り込まれた決定リスト達の記述量を比較し、最小値を求め、第２記憶装置56から記述量を最小になる決定リストを出力させることを指示する制御信号を送ることを指令する信号を、第２制御信号発生回路58に送り込む。58は57の指令を受けて、第２記憶装置56から記述量を最小にする決定リストを出力させることを指示する制御信号を第２記憶装置56に送る。第２記憶装置56は58の指令を受けて、記述量を最小にする決定リストを出力する。

第６図は発明４の決定リストの発生装置の実施例を示すブロック図である。観測データを装置全体の入力として、該入力は先ず、DL＊（ｋ）発生回路61に送り込まれる。61は固定されたｋに対して、発明３の装置と同じ回路によって決定リストを発生させ（固定されたｋに対して発生された決定リストをDL＊（ｋ）とする）、これを第１制御信号発生装置62より発生する制御信号の指示に従って複数のｋの値に対して実行する。61は各ｋに対するDL＊（ｋ）と各決定に用いられたタームのそれぞれに対するA^+（ｔ），A^-（ｔ）の値を順に記述量計算回路63に送り込み、各ｋに対するDL＊（ｋ）を記憶装置64に送り込む。63は61の出力を入力として、それぞれの記述量を計算し、その記述量を記述量比較回路65及び記憶装置64に送り込む。ここで、記述量は、決定リスト自身の記述量と決定に対する例外データの記述量との単純な和ではなく、重み付き和、すなわち、λ（決定リストの記述量）＋（１－λ）（例外データの記述量）（０＜λ＜１）として計算する場合、あるいは決定リストの記述量と例外データの記述量のいずれかを補正して結合した値を用いる場合もあるとする。記憶装置64は61及び63の出力を入力として、これらを記憶する。65は63の出力を入力として、65は各ｋに対するDL＊（ｋ）の記述量を比較し、最小値を求め、記憶装置64から記述量を最小にする決定リストを出力させることを指示する制御信号を送ることを指令する信号を、第２制御信号発生回路66に送り込む。66は65の指令を受けて、記憶装置64から記述量を最小にする決定リストを出力させることを指示する制御信号を記憶装置64に送る。記憶装置64は66の指令を受けて、記述量を最小にする決定リストを出力する。

（発明の効果）

発明１及び発明３によれば、固定されたｋの値（決定を規定する論理積に現れる文字の最大数）に対して、観測データと同じ情報源から発生する未知データを高い制度で分類予測することが理論的に保証された決定リストが発生できる。発明２及び発明４によれば、さらに、ｋの値を動かして得られる広い決定リストの集合の中から、観測データと同じ情報源から発生する未知データを高い精度で分類予測することが理論的に保証され、また観測データをコンパクトに記述できる決定リストを選ぶことが出来る。

【図面の簡単な説明】

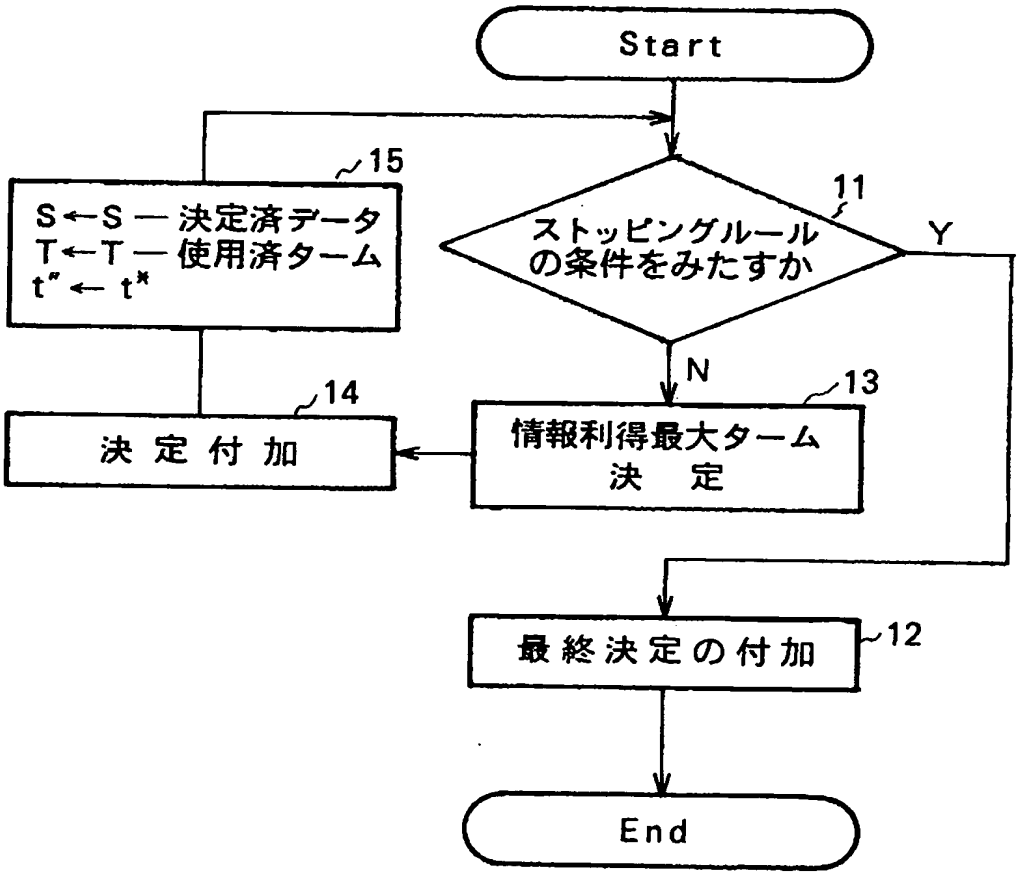
第１図は発明１の生成方法を示すフローチャート、第２図は発明１の決定リストの生成方法で、第１図に示した方法の変形版のフローチャート、第３図は発明２の決定リストの生成方法を示すフローチャート、第４図は発明

３の決定リストの生成装置を示すブロック図、第５図は発明３の決定リストの生成装置で、第４図に示した方法の変形版のブロック図、第６図は発明４の決定リストの生成装置を示すブロック、第７図、第８図、第９図は本発明の作用を説明するための図である。

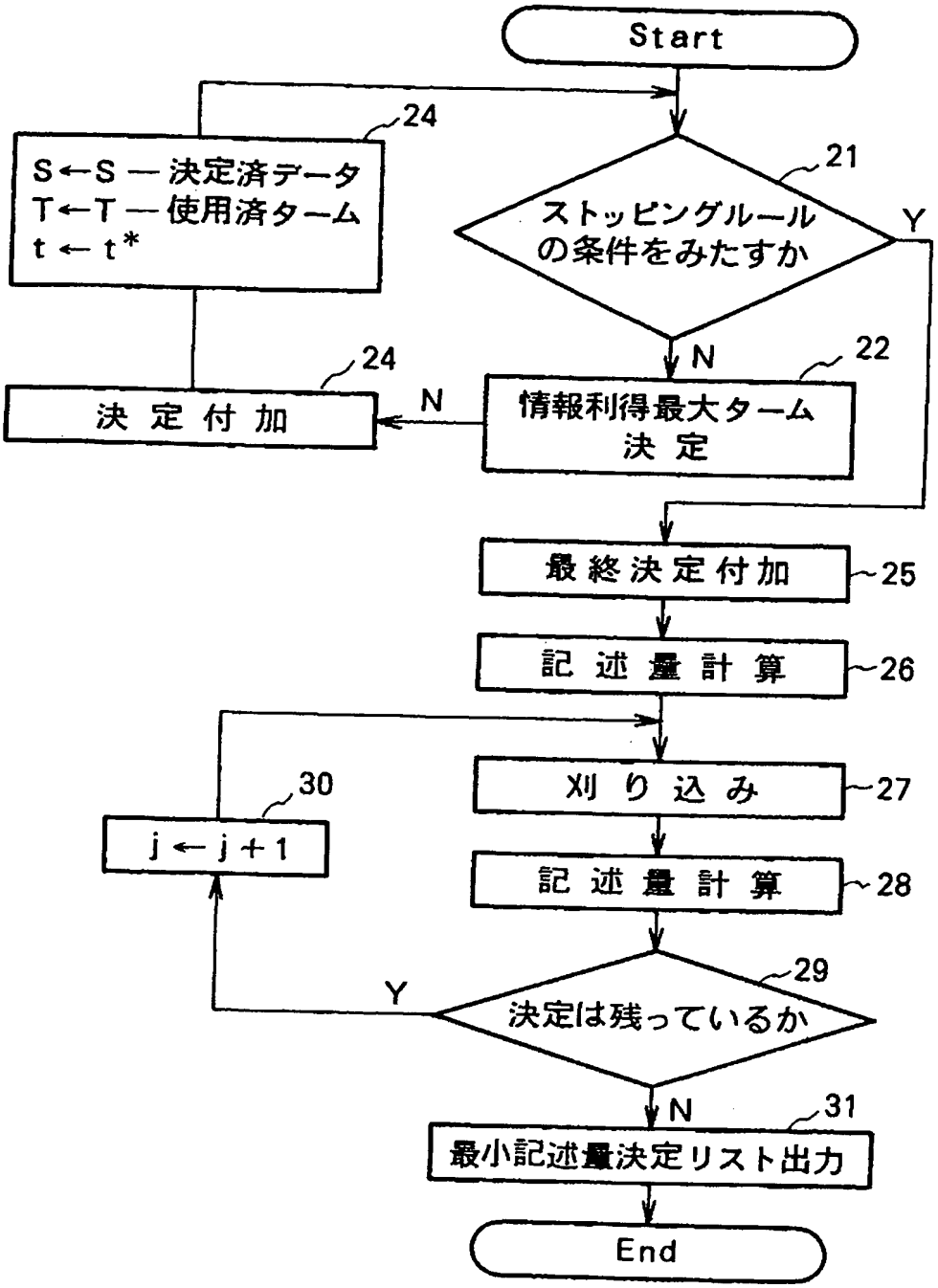
図において、

41:記憶装置、42:情報利得最大ターム決定回路、43:決定付加回路、44:制御信号発生装置、51:第１記憶装置、52:情報利得最大ターム決定回路、53:決定付加回路、54:第１制御信号発生装置、55:刈り込み及び記述量計算回路、56:第２記憶装置、57:記述量比較回路、58:第２制御信号発生装置、61:DL* （ｋ）発生回路、62:第１制御信号発生装置、63:記述量計算回路、64:記憶装置、65:記述量比較回路、66:第２制御信号発生装置。

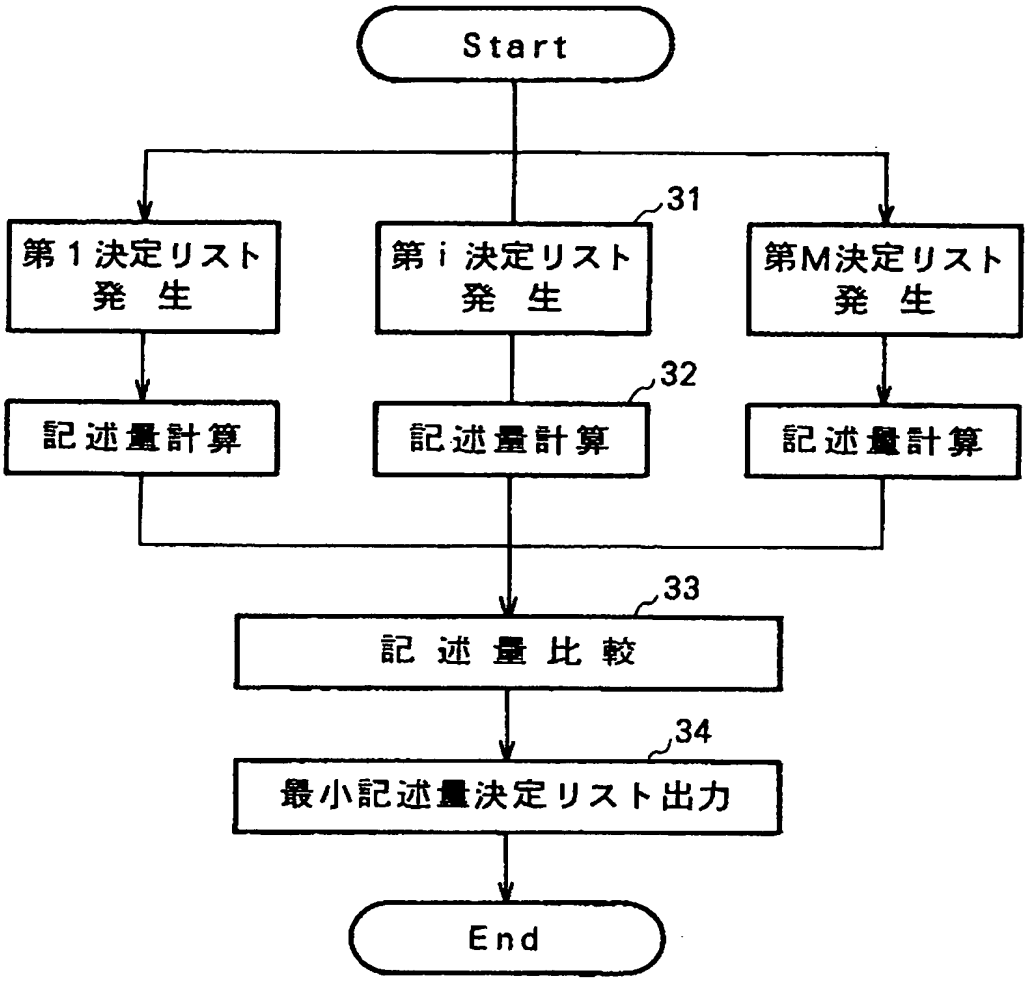
【第１図】



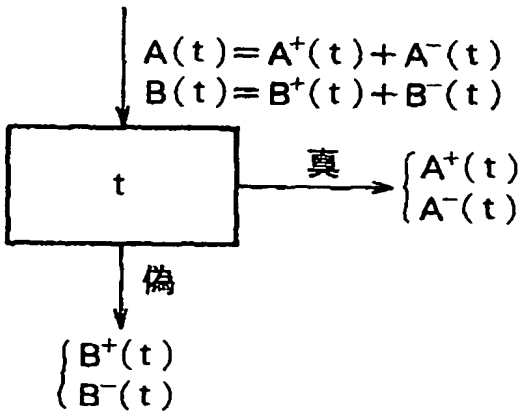
【第２図】


Start
21
ストッピングルールの条件をみたすか
Y
N
22
情報利得最大ターム決定
N
24
決定付加
24
S←S－決定済データ
T←T－使用済ターム
t←t*
最終決定付加
25
記述量計算
26
刈り込み
27
記述量計算
28
決定は残っているか
29
Y
30
j←j+1
N
31
最小記述量決定リスト出力
End

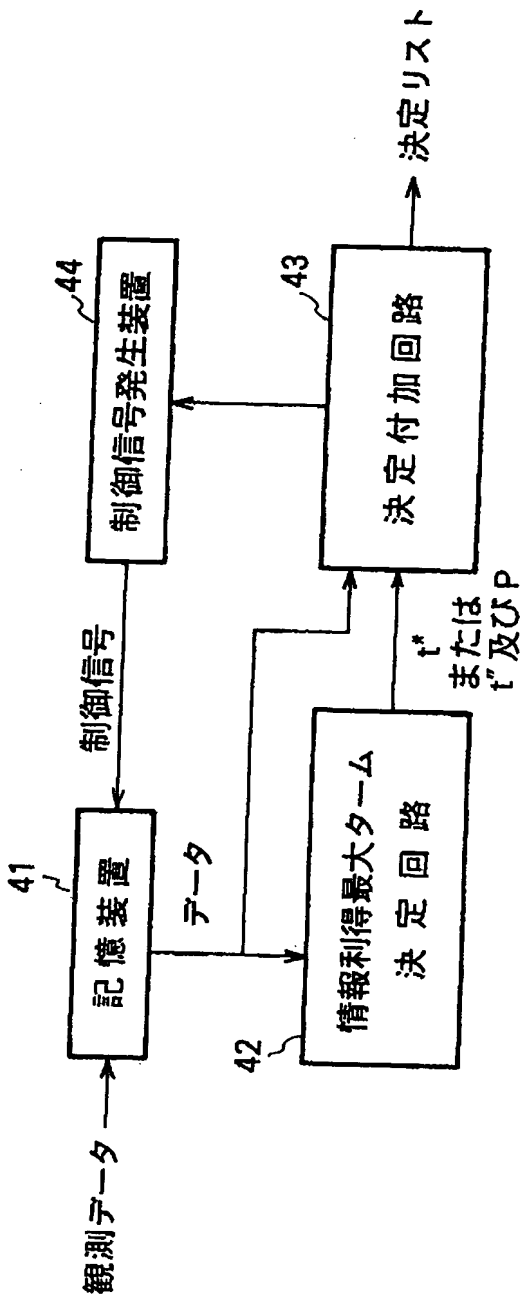
【第３図】


Start
第１決定リスト発生
第ｉ決定リスト発生
第M決定リスト発生
31
記述量計算
記述量計算
記述量計算
32
記述量比較
33
最小記述量決定リスト出力
34
End

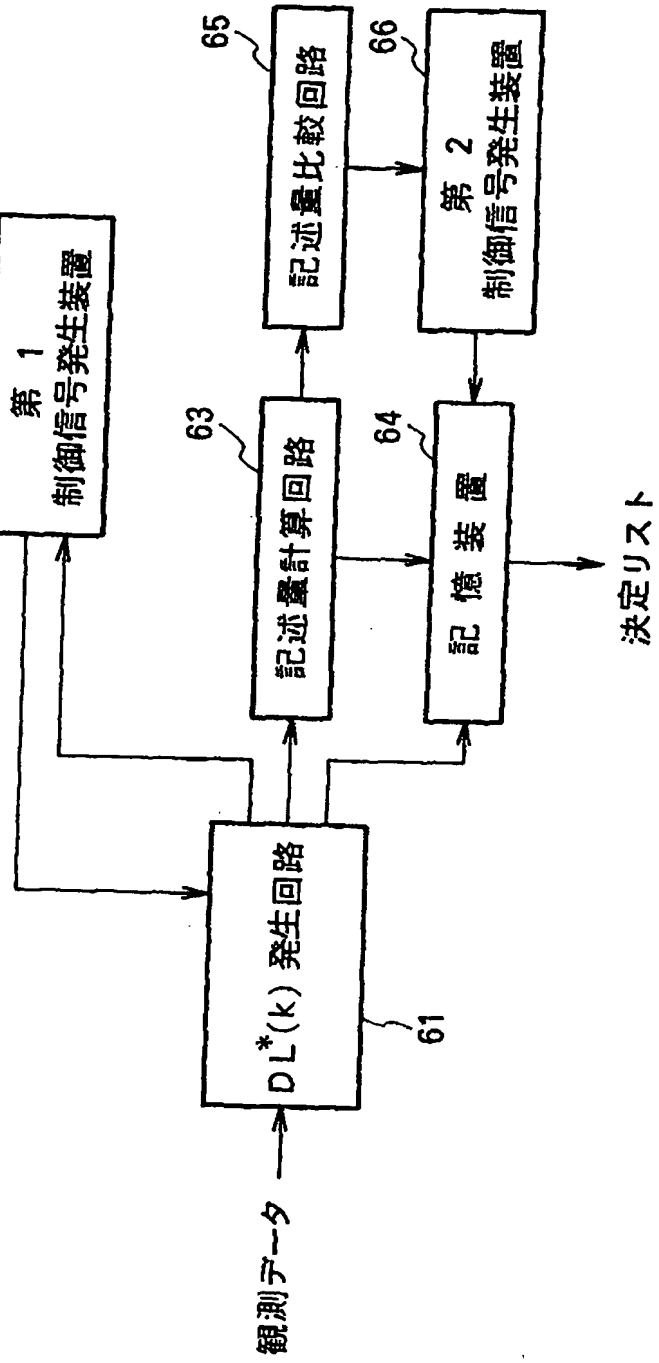
【第８図】


A(t)＝A⁺(t)＋A⁻(t)
B(t)＝B⁺(t)＋B⁻(t)
t
真
A⁺(t)
A⁻(t)
偽
B⁺(t)
B⁻(t)

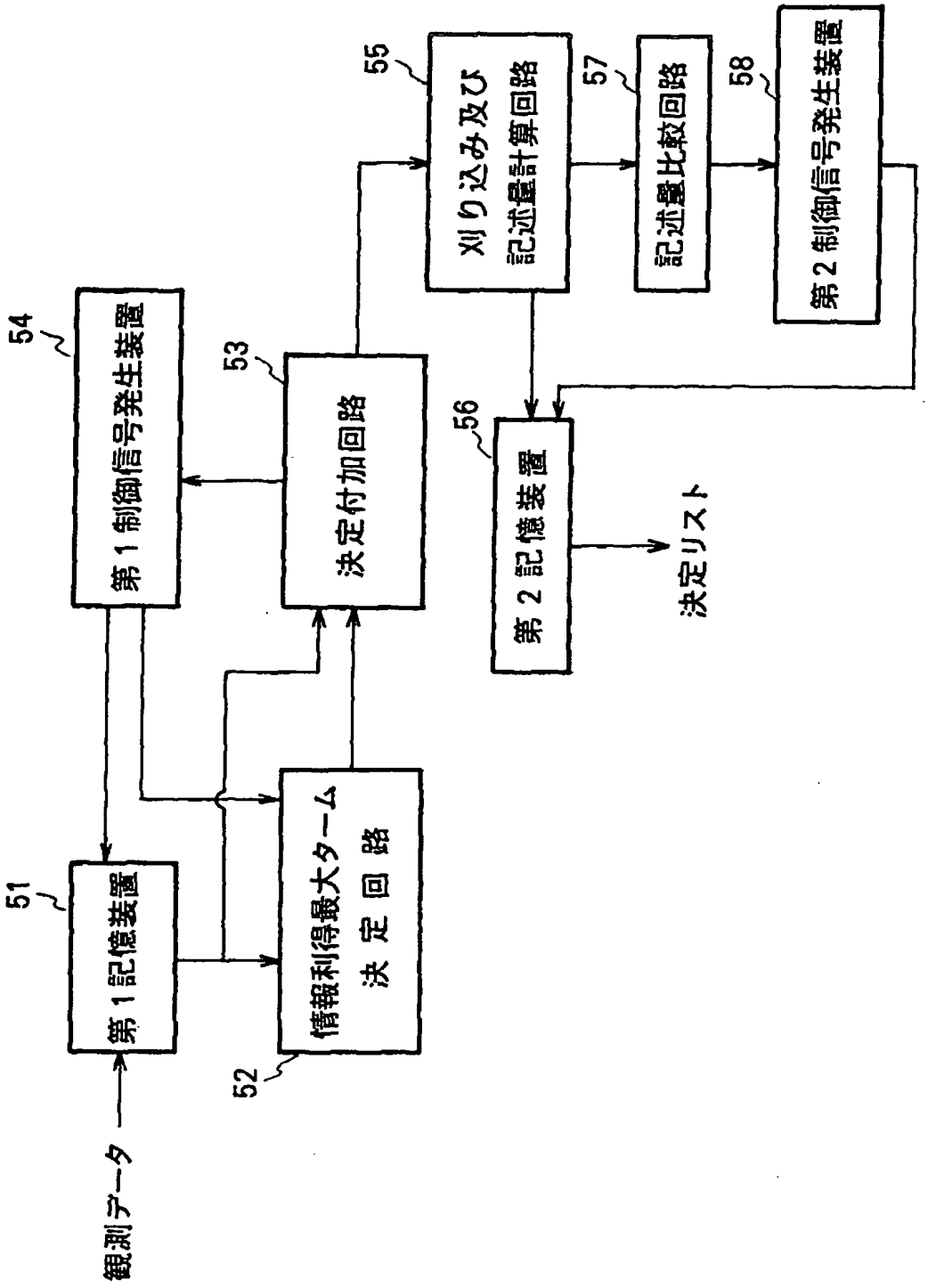
【第４図】


観測データ
41
記憶装置
データ
制御信号
44
制御信号発生装置
42
情報利得最大ターム
決定回路
t*
または
t*及びP
43
決定付加回路
決定リスト

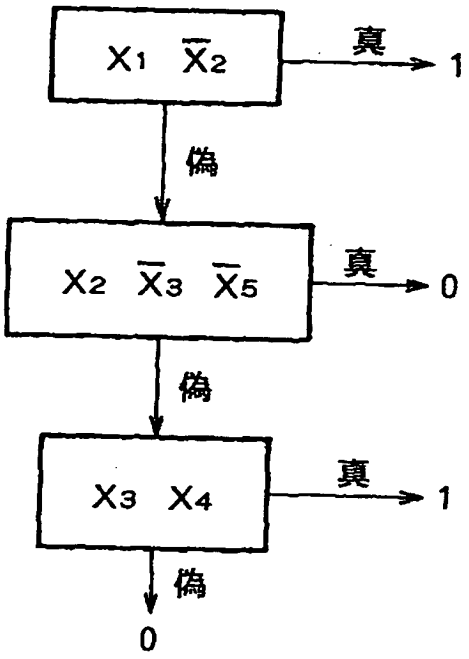
【第６図】


観測データ
61
DL*(k)発生回路
62
第1
制御信号発生装置
63
記述量計算回路
64
記憶装置
65
記述量比較回路
66
第2
制御信号発生装置
決定リスト

【第５図】


観測データ
第１記憶装置
51
情報利得最大ターム決定回路
52
決定付加回路
53
第１制御信号発生装置
54
刈り込み及び記述量計算回路
55
第２記憶装置
56
記述量比較回路
57
第２制御信号発生装置
58
決定リスト

【第７図】



【第９図（ａ）】

対象	x_1	x_2	x_3	x_4	x_5	x_6	クラス
1	0	1	1	0	1	1	1
2	1	1	1	0	0	0	1
3	0	1	0	1	1	0	1
4	0	1	1	0	0	1	1
5	1	1	1	0	1	1	1
6	1	0	0	0	0	1	0
7	0	1	1	1	1	0	1
8	1	0	1	0	1	1	0
9	0	1	0	1	1	0	0
10	1	1	1	0	1	1	1
11	0	1	1	0	0	1	1
12	0	1	1	1	0	0	0
13	1	0	0	0	0	0	0
14	0	1	1	1	1	1	1
15	0	1	1	0	1	1	1
16	1	1	1	0	0	1	0
17	0	1	1	0	1	1	1
18	0	1	0	1	0	1	0

【第９図（ｂ）】

対象	x_1	x_2	x_3	x_4	x_5	x_6	クラス
19	1	0	1	0	0	0	0
20	0	1	1	1	0	0	1
21	0	1	1	0	0	0	1
22	0	1	0	0	0	0	0
23	1	1	1	1	1	1	1
24	1	0	1	0	0	0	1
25	0	1	0	1	1	0	1
26	1	0	1	1	0	0	1
27	0	1	1	0	0	1	1
28	1	0	0	0	0	1	0
29	0	1	1	0	0	0	1
30	0	1	0	1	1	0	1
31	0	0	0	0	0	1	0
32	1	1	1	1	1	0	0
33	1	0	1	0	1	1	0
34	1	1	0	1	1	0	0
35	1	0	1	0	1	1	0

Under Condition 14, according to [11] and [13], we define the *generalized inverse function* of L_0 by $L_0^{-1}(L_0(Z)) = Z$ for $0 \le Z \le 1$ and $L_0^{-1}(Z) \ge 1$ for $Z \ge L_0(1)$. Similarly, we define the generalized inverse function of L_1 by $L_1^{-1}(L_1(Z)) = Z$ for $0 \le Z \le 1$ and $L_1^{-1}(Z) \le 0$ for $Z \ge L_1(0)$.

Example 5: For the entropic loss function

$$L(Y, Z) = Y \ln(Y/Z) + (1 - Y) \ln((1 - Y)/(1 - Z))$$

we see that $L_0(Z) = -\ln(1 - Z)$ and $L_1(Z) = -\ln Z$. Then we have $L_0^{-1}(Z) = 1 - e^{-Z}$ and $L_1^{-1}(Z) = e^{-Z}$. A simple calculation yields $\lambda^* = 1$ (see [11]). We see that Conditions 14–16 are satisfied for L.

Example 6: For the quadratic loss function $L(Y, Z) = (Y - Z)^2$, we see that $L_0(Z) = Z^2$ and $L_1(Z) = (1 - Z)^2$. Then we have $L_0^{-1}(Z) = \sqrt{Z}$ and $L_1^{-1}(Z) = 1 - \sqrt{Z}$. A simple calculation yields $\lambda^* = 2$ (see [11]). We see that Conditions 14–16 are satisfied for L.

Example 7: For the Hellinger loss function

$$L(Y, Z) = \frac{1}{2}\left((\sqrt{Y} - \sqrt{Z})^2 + (\sqrt{1 - Y} - \sqrt{1 - Z})^2\right)$$

we see that $L_0(Z) = 1 - \sqrt{1 - Z}$ and $L_1(Z) = 1 - \sqrt{Z}$. Then we can set

$$L_0^{-1}(Z) = 2Z - Z^2 (0 \le Z \le 1),\ L_0^{-1}(Z) = 1 (Z \ge 1)$$

and

$$L_1^{-1}(Z) = (1 - Z)^2 (0 \le Z \le 1),\ L_1^{-1}(Z) = 0 (Z \ge 1).$$

A simple calculation shows $\lambda^* = \sqrt{2}$ (see [11]). We see that Conditions 14 and 15 are satisfied, but Condition 16 is not satisfied.

Example 8: For the logistic loss function as in (3), we see that

$$L_0(Z) = (1/2) \ln((e^{2(2Z-1)} + 1)/(1 + e^{-2}))$$

and

$$L_1(Z) = (1/2) \ln((1 + e^{-2(2Z-1)})/(1 + e^{-2})).$$

Then we have

$$L_0^{-1}(Z) = 1/2 + (1/4) \ln((1 + e^{-2})e^{2Z} - 1)$$

and

$$L_1^{-1}(Z) = 1/2 - (1/4) \ln((1 + e^{-2})e^{2Z} - 1).$$

A simple calculation yields $\lambda^* = 2$. We see that Conditions 14–16 are satisfied for L.

Under the above notation, a variant of Kivinen and Warmuth's version of the aggregating strategy for the case of $\mathcal{Y} = [0, 1]$, which we denote by AGG, is described as follows:

Algorithm AGG: Let $\mathcal{H}_k$, L, π, and $\lambda > 0$ be given.

At each time t, on receiving X_t, compute

$$\Delta_t(0) \stackrel{\text{def}}{=} L(0|D^{t-1}, X_t)$$

and

$$\Delta_t(1) \stackrel{\text{def}}{=} L(1|D^{t-1}, X_t)$$

where the notation of $L(Y|D^{t-1}, X_t)(Y = 0, 1)$ follows (21). Then predict Y_t with any value $\hat{Y}_t$ satisfying

$$L_1^{-1}(\Delta_t(1)) \le \hat{Y}_t \le L_0^{-1}(\Delta_t(0)). \tag{26}$$

If no such $\hat{Y}_t$ exists, the algorithm fails. We write $\hat{Y}_t = \mathrm{AGG}_t(X_t)$.

After prediction, the correct outcome Y_t is received.

Note that if $L_1^{-1}(\Delta_t(1)) \le L_0^{-1}(\Delta_t(0))$ holds, then $\hat{Y}_t$ satisfying (26) is given by, for example,

$$\hat{Y}_t = \frac{1}{2}(L_1^{-1}(\Delta_t(1)) + L_0^{-1}(\Delta_t(0))).$$

Note also that (21) can be written as

$$L(Y|D^{t-1}, X_t) = -\frac{1}{\lambda} \ln \int d\theta v_t(\theta) \exp(-\lambda L((X_t, Y) : f_\theta))$$

where $v_t(\theta) = w_t(\theta)/\int w_t(\theta)\, d\theta$ and $w_t(\theta)$ follows the following updating rule:

$$w_t(\theta) = w_{t-1}(\theta) \exp(-\lambda L(D_{t-1} : f_\theta))$$

which implies that $w_t(\theta)$ *multiplicatively decays* at each time and that $\Delta_t(0)$ and $\Delta_t(1)$ in AGG can be computed incrementally with respect to t.

We say that L is λ-*realizable* when AGG never fails for λ. Haussler *et al.* [11] proved that in the case of $\mathcal{Y} = \{0, 1\}$, under Conditions 14 and 15, L is λ-realizable if and only if $\lambda \le \lambda^*$ for λ^* as in (25).

Kivinen and Warmuth [13] proved that if L satisfies Condition 16, then for $\hat{Y}_t = \mathrm{AGG}_t(X_t)$ satisfying (26), for all $Y_t \in [0, 1]$, the following inequality holds:

$$L((X_t, Y_t) : \mathrm{AGG}_t) = L(D_t : \mathrm{AGG}_t) \le L(Y_t|D^{t-1}, X_t). \tag{27}$$

This implies that $\hat{Y}_t$ satisfying (26) is well-defined in the sense that the loss for AGG with respect to any real outcome Y_t in $[0, 1]$ is upper-bounded by $L(Y_t|D^{t-1}, X_t)$.

C. Cumulative Loss Bounds for the Aggregating Strategy

Below we give upper bounds on the cumulative loss for AGG using $\mathcal{H}_k$. First note that summing both sides of (27) with respect to t gives

$$\sum_{t=1}^{m} L(D_t : \mathrm{AGG}_t) \le \sum_{t=1}^{m} L(Y_t|D^{t-1}, X_t) = I(D^m : \mathcal{H}_k).$$

Here we have used Lemma 1 to derive the last equation. Combining this inequality with (4) and (5) immediately gives the following theorem.

Theorem 2. Upper Bounds on Cumulative Loss for AGG: Let $\mathcal{Y} = [0, 1]$ and suppose that the loss function L satisfies Conditions 14–16. Let λ^* be the largest number such that L is λ^*-realizable, i.e., λ^* is as in (25). Let P be the target distribution. Under Conditions 1–7 for P, $\mathcal{H}_k$, L, and π, the

expected cumulative loss for AGG with $\lambda = \lambda^*$ using $\mathcal{H}_k$ is upper-bounded as follows:

$$\limsup_{m\to\infty}\left(E\left[\sum_{t=1}^{m} L(D_t : \mathrm{AGG}_t)\right] - \left(mE[L(D : f_{\theta_0})] + \frac{k}{2\lambda^*}\ln\frac{m\lambda^*}{2\pi}\right)\right) \le \frac{1}{\lambda^*}\ln\frac{\sqrt{|J(\theta_0)|}}{\pi(\theta_0)} \quad (28)$$

where the notation of θ_0 and $|J(\theta_0)|$ follows Conditions 3 and 5, respectively.

Under Conditions 1 and 8–12 for $\mathcal{H}_k$, L,, and π, for all $D^m \in \mathcal{D}^m(\Theta_k)$, the cumulative loss for AGG with $\lambda = \lambda^*$ using $\mathcal{H}_k$ with respect to D^m is upper-bounded as follows:

$$\sum_{t=1}^{m} L(D_t : \mathrm{AGG}_t) \le \sum_{t=1}^{m} L(D_t : f_{\hat{\theta}}) + \frac{k}{2\lambda^*}\ln\frac{m\lambda^*\mu}{2\pi} + \frac{1}{\lambda^*}\ln\frac{1}{r\underline{c}} + o(1) \quad (29)$$

where $o(1)$ goes to zero uniformly in $D^m \in \mathcal{D}^m(\Theta_k)$ as m goes to infinity. The notation of $\mathcal{D}^m(\Theta_k)$, $\hat{\theta}$, μ, r, and $\underline{c}$ follows Conditions 9–12.

For the case of $\mathcal{Y} = \{0, 1\}$, (28) and (29) hold under Conditions 14 and 15 only (i.e., without Condition 16).

Define the minimum expected cumulative loss over $\mathcal{H}_k$ by

$$E\left[\sum_{t=1}^{m} L(D_t : f_{\theta_0})\right] = mE[L(D : f_{\theta_0})]$$

which cannot be attained without knowing P. Equation (28) implies that the expected cumulative loss for AGG is within

$$\frac{k}{2\lambda^*}\ln\frac{m\lambda^*}{2\pi} + \frac{1}{\lambda^*}\ln\frac{\sqrt{|J(\theta_0)|}}{\pi(\theta_0)} + o(1)$$

of the minimum expected cumulative loss over $\mathcal{H}_k$. Further define the minimum empirical cumulative loss over $\mathcal{H}_k$ by

$$\min_{\theta\in\Theta_k}\sum_{t=1}^{m} L(D_t : f_\theta)$$

which is attainable using a single $\theta \in \Theta_k$ but cannot be attained by any on-line prediction algorithm for most sequences. Equation (29) implies that the worst case cumulative loss for AGG is within

$$\frac{k}{2\lambda^*}\ln\frac{m\lambda^*\mu}{2\pi} + \frac{1}{\lambda^*}\ln\frac{1}{r\underline{c}} + o(1)$$

of the minimum empirical cumulative loss over $\mathcal{H}_k$, where the worst case is taken with respect to D^m over $\mathcal{D}^m(\Theta_k)$.

Corresponding to (18) and (19), we are able to derive tighter upper bounds on the expected cumulative loss and the probabilistic cumulative loss for AGG. We omit the argument here.

Finally, note that in the case where the hypothesis class is a union set $\mathcal{H} = \cup_{k=1}^{s}\mathcal{H}_k$ $(s < \infty)$ with respect to k instead of $\mathcal{H}_k$ for a single k, under the conditions as in Theorem 2, for any $D^m \in \cap_{k=1}^{s}\mathcal{D}^m(\Theta_k)$, we can upper-bound the worst case cumulative loss for AGG using $\mathcal{H}$ by the following quantity, which is an upper bound on the ESC of the form of (2):

$$\min_{1\le k\le s}\left\{\sum_{t=1}^{m} L(D_t : f_{\hat{\theta}}) + \frac{k}{2\lambda^*}\ln\frac{m\lambda^*\mu}{2\pi} + \frac{1}{\lambda^*}\ln\frac{1}{r\underline{c}} - \ln\pi(k)\right\} + o(1)$$

where λ^* is as in (25).

Example 9: For $\mathcal{H}_k$, L, and π as in Example 1, if the target distribution P satisfies Conditions 2–6, the expected cumulative loss for AGG for sample size m is upper-bounded by

$$mE[(Y - \theta_0^T X)^2] + \frac{k}{4}\ln\frac{m}{\pi} + \frac{1}{2}\ln\frac{\pi^{k/2}\sqrt{|J_0|}}{2^k\Gamma(1 + k/2)} + o(1)$$

where J_0 is the matrix for which the (i, j)th component is $E_P[X^{(i)}X^{(j)}]$ and $\theta_0 = (J_0)^{-1}(E_P[YX])$.

For given D^m, let

$$\hat{\theta} = (\hat{J})^{-1}\left(\frac{1}{m}\sum_{t=1}^{m} Y_t X_t\right)$$

where $\hat{J}$ is a matrix for which the (i, j)th component is given by $\frac{1}{m}\sum_{t=1}^{m} X_t^{(i)}X_t^{(j)}$. Let $\mathcal{D}^m(\Theta_k) = \{D^m \in \mathcal{D}^m : \hat{J}$ is regular and $\hat{\theta} \in \Theta_k\}$. Then setting $\lambda^* = 2$, $\mu = 2k$, $r = 1/2^k$, and $\underline{c} = 2^k\Gamma(1 + k/2)/\pi^{k/2}$, for all $D^m \in \mathcal{D}^m(\Theta_k)$, the cumulative loss for AGG with respect to D^m is upper-bounded by

$$\sum_{t=1}^{m}(Y_t - \hat{\theta}^T X_t)^2 + \frac{k}{4}\ln\frac{2mk}{\pi} + \frac{1}{2}\ln\frac{\pi^{k/2}}{\Gamma(1 + k/2)} + o(1).$$

Example 10: For $\mathcal{H}$, L, and π as in Example 2, if the target distribution P satisfies Conditions 2–6, the expected cumulative loss for AGG for sample size m is upper-bounded by

$$mE[L(D : f_{\theta_0})] + \frac{1}{4}\ln\frac{m}{\pi} + \frac{1}{4}\ln J_0 + o(1),$$

where

$$J_0 = \frac{8}{(e^{2\theta_0 - 1} + e^{-2\theta_0 + 1})^2}$$

and

$$\theta_0 = \frac{1}{2} + \frac{1}{4}\ln\frac{1 + E_P[2Y - 1]}{1 - E_P[2Y - 1]}.$$

For given D^m, let

$$\hat{\theta} = \frac{1}{2} + \frac{1}{4}\ln\frac{1 + \frac{1}{m}\sum_{t=1}^{m}(2Y_t - 1)}{1 - \frac{1}{m}\sum_{t=1}^{m}(2Y_t - 1)}.$$

Let $\mathcal{D}^m(\Theta) = \{D^m \in \mathcal{D}^m : \hat{\theta} \in \Theta\}$. Then setting $k = 1$, $\lambda^* = 2$, $\mu = 2$, $r = 1/2$, and $\underline{c} = 1$, for all $D^m \in \mathcal{D}^m(\Theta)$, the cumulative loss for AGG with respect

to D^m is upper-bounded by

$$\sum_{t=1}^{m} L(D_t : f_{\hat{\theta}}) + \frac{1}{4} \ln \frac{2m}{\pi} + \frac{1}{2} \ln 2 + o(1).$$

Example 11: For $\mathcal{H}$, L, and π as in Example 3, for given D^m, let

$$\hat{\theta} = \frac{\left(\sum_{t=1}^{m} \sqrt{Y_t}\right)^2}{\left(\sum_{t=1}^{m} \sqrt{Y_t}\right)^2 + \left(\sum_{t=1}^{m} \sqrt{1 - Y_t}\right)^2}.$$

Let $\mathcal{D}^m(\Theta) = \{D^m \in \mathcal{D}^m : \hat{\theta} \in \Theta\}$. Then setting $k = 1$, $\lambda^* = \sqrt{2}$, $\mu = (1/2)F(\varepsilon)$, $r = 1/2$, and $\underline{c} = 1/(1 - 2\varepsilon)$, for all $D^m \in \mathcal{D}^m(\Theta)$, the cumulative loss for AGG with respect to D^m is upper-bounded by

$$\sum_{t=1}^{m} L(D_t : f_{\hat{\theta}}) + \frac{1}{2\sqrt{2}} \ln \frac{\sqrt{2} m F(\varepsilon)}{4\pi} + \frac{1}{\sqrt{2}} \ln 2(1-2\varepsilon) + o(1).$$

Example 12: Consider $\mathcal{H}$ as in Example 4 as a class of real-valued functions through $f_\theta(X) = f_\theta(1|X)$. Then $\mathcal{H}$, L, and π as in Example 4, setting $\lambda^* = 2$, $\mu = 2$, $r = 1/2^k$, and $\underline{c} = 1$, for all $D^m \in \mathcal{D}^m$, the cumulative loss for AGG with respect to D^m is upper-bounded by

$$\sum_{i=1}^{k} \frac{m_{1i}(m - m_{1i})}{m_i} + \frac{k}{4} \ln \frac{2m}{\pi} + \frac{k}{2} \ln 2 + o(1)$$

where m_i is the number of examples for which X fell into $S_i (i = 1, \cdots, k)$ and m_{1i} is the number of examples for which $Y = 1$ and X fell into $S_i (i = 1, \cdots, k)$.

IV. Applications of ESC to Batch-Learning

A. Batch-Learning Model

In this section we consider an application of ESC to the design and analysis of a batch-learning algorithm. In general, a *batch-learning algorithm* (see, e.g., [10]) is an algorithm that takes as input a sequence of examples: $D^m = D_1 \cdots D_m \in \mathcal{D}^*$ $(D_t = (X_t, Y_t), t = 1, \cdots, m)$ and a hypothesis class $\mathcal{H}$, and then outputs a single hypothesis belonging to $\mathcal{H}$.

Let $\mathcal{F}$ be a set of all functions from $\mathcal{X}$ to $\mathcal{Z}$ in the case where $\mathcal{Z} \subset \boldsymbol{R}$, or let $\mathcal{F}$ be a set of all conditional probability densities (or probability mass functions) over $\mathcal{Y}$ for given $X \in \mathcal{X}$ in the case where $\mathcal{Z}$ is a set of conditional probability densities (or probability mass functions) over $\mathcal{Y}$ for given $X \in \mathcal{X}$. Suppose that each D is independently drawn according to the target distribution P over $\mathcal{D}$. For a hypothesis $f \in \mathcal{H}$, we define a *generalization loss* of f with respect to P by

$$\Delta_P(f) \stackrel{\text{def}}{=} E_P[L(D : f)] - \inf_{h \in \mathcal{F}} E_P[L(D : h)]$$

where E_P denotes the expectation taken for the generation of $D = (X, Y)$ with respect to P. For sample size m, we also define a *statistical risk* for a batch-learning algorithm $\mathcal{A}$ as the expected value of the generalization loss:

$$E[\Delta_P(\hat{f})]$$

where $\hat{f}$ is an output of $\mathcal{A}$, which is a random variable depending on the input sequence D^m, and the expectation E is taken for the generation of D^m with respect to $P(D^m)$. Our goal is to design a batch-learning algorithm for which the statistical risk is as small as possible.

B. The Minimum L-Complexity Algorithm

Next we consider a batch-approximation of ESC by a single hypothesis to motivate a batch-learning algorithm. We now approximate the integral in (1) by quantizing Θ_k. For a k-dimensional parametric hypothesis class $\mathcal{H}_k = \{f_\theta : \theta \in \Theta_k \subset \boldsymbol{R}^k\}$, let $\Theta_k^{(m)}$ be a finite subset of Θ_k depending on sample size m. We define $\mathcal{H}_k^{(m)} (\subset \mathcal{H}_k)$ by $\mathcal{H}_k^{(m)} \stackrel{\text{def}}{=} \{f_\theta : \theta \in \Theta_k^{(m)}\}$. We refer to $\Theta_k^{(m)}$ as a *quantization* of Θ_k. Similarly, we refer to $\mathcal{H}_k^{(m)}$ as a *quantization* of $\mathcal{H}_k$. We call a map $\tau_m : \Theta_k \to \Theta_k^{(m)}$ a *truncation*. Similarly, we also call a map $\mathcal{H}_k \to \mathcal{H}_k^{(m)}$ defined by $f_\theta \in \mathcal{H}_k \to f_{\tau_m(\theta)} \in \mathcal{H}_k^{(m)}$ a *truncation* of f_θ. For $\theta \in \Theta_k^{(m)}$, let $S(\theta) \stackrel{\text{def}}{=} \{\theta' \in \Theta_k : \tau_m(\theta') = \theta\}$, which is a set of real-valued points which are truncated to θ by τ_m.

Choosing τ_m so that for each $\theta \in \Theta_k^{(m)}$, the Lebesgue measure of $S(\theta)$ goes to zero as m increases to infinity, we may consider an approximation of ESC by the quantity $J(D^m : \mathcal{H}_k)$ defined as follows:

$$J(D^m : \mathcal{H}_k) \stackrel{\text{def}}{=} -\frac{1}{\lambda} \ln \sum_{\theta \in \Theta_k^{(m)}} W(\theta) \cdot \exp\left(-\lambda \sum_{t=1}^{m} L(D_t : f_\theta)\right)$$

where $W(\theta) \stackrel{\text{def}}{=} \int_{\theta' \in S(\theta)} \pi(\theta')\, d\theta'$ for a given prior density $\pi(\theta)$.

Since

$$\sum_{\theta \in \Theta_k^{(m)}} W(\theta) \exp\left(-\lambda \sum_{t=1}^{m} L(D_t : f_\theta)\right) \geq \max_{\theta \in \Theta_k^{(m)}} W(\theta) \exp\left(-\lambda \sum_{t=1}^{m} L(D_t : f_\theta)\right)$$

the quantity $J(D^m : \mathcal{H}_k)$ can be upper-bounded as follows:

$$\begin{aligned} J(D^m : \mathcal{H}_k) &\leq -\frac{1}{\lambda} \ln \max_{\theta \in \Theta_k^{(m)}} W(\theta) \exp\left(-\lambda \sum_{t=1}^{m} L(D_t : f_\theta)\right) \\ &= \min_{\theta \in \Theta_k^{(m)}} \left\{\sum_{t=1}^{m} L(D_t : f_\theta) - \frac{1}{\lambda} \ln W(\theta)\right\}. \end{aligned}$$

Notice here that $W(\theta)$ can be thought of as a probability mass of θ over $\Theta_k^{(m)}$ since

$$\sum_{\theta \in \Theta_k^{(m)}} W(\theta) = 1.$$

Hence $-\ln W(\theta)$ can be interpreted as the codelength for θ. This implies that letting L_m be any function: $\Theta_k^{(m)} \to \boldsymbol{R}^+$

satisfying Kraft's inequality:

$$\sum_{\theta \in \Theta_k^{(m)}} e^{-L_m(\theta)} \leq 1$$

we can upper-bound $J(D^m : \mathcal{H}_k)$ by

$$\min_{\theta \in \Theta_k^{(m)}} \left\{ \sum_{t=1}^{m} L(D_t : f_\theta) + \frac{1}{\lambda} L_m(\theta) \right\}. \tag{30}$$

This argument can be easily extended to a batch-approximation of ESC relative to a union set $\mathcal{H} = \cup_k \mathcal{H}_k$ with respect to k. That is, ESC of the form of (2) can be approximated by

$$\min_k \min_{\theta \in \Theta_k^{(m)}} \left\{ \sum_{t=1}^{m} L(D_t : f_\theta) + \frac{1}{\lambda} L_m(\theta, k) \right\} \tag{31}$$

where $L_m(\cdot, \cdot)$ is a function $\cup_k \Theta_k^{(m)} \times \{1, 2, \cdots\} \rightarrow \boldsymbol{R}^+$ satisfying

$$\sum_k \sum_{\theta \in \Theta_k^{(m)}} e^{-L_m(\theta, k)} \leq 1.$$

From the above discussion we see that the best batch-approximation of ESC can be realized by a single hypothesis that minimizes the weighted sum of the empirical loss for the hypothesis with respect to D^m and the codelength for the hypothesis. This fact motivates a batch-learning algorithm that produces from a data sequence D^m a hypothesis that attains the minimum of (31). For a loss function L, we name this algorithm the *minimum L-complexity algorithm*, which we denote by MLC.

In order to define MLC, we have to fix a method of quantization for Θ_k. A question arises in how finely we should quantize a continuous parameter space to approximate ESC best. The optimal quantization scale can be obtained similarly with the argument by Rissanen [19, pp. 55–56] as follows: Let $\delta = (\delta_1, \cdots, \delta_m)$ be the maximal quantization scale around the truncated value $\tau_m(\hat{\theta})$ of $\hat{\theta}$ where

$$\hat{\theta} = \arg \min_{\theta \in \Theta_k} \sum_{t=1}^{m} L(D_t : f_\theta)$$

with

$$\partial \sum_{t=1}^{m} L(D_t : f_\theta) / \partial \theta|_{\theta = \hat{\theta}} = 0.$$

Applying Taylor's expansion of $L(D_t : f_{\hat{\theta}+\delta})$ around $\hat{\theta}$ up to the second order, we have

$$\sum_{t=1}^{m} L(D_t : f_{\hat{\theta}+\delta}) = \sum_{t=1}^{m} L(D_t : f_{\hat{\theta}}) + \frac{m}{2} \delta^T \Sigma \delta + mo(\delta^2)$$

where

$$\Sigma = \left(\frac{1}{m} \frac{\partial^2 \sum_{t=1}^{m} L(D_t : f_\theta)}{(\partial \theta_i \partial \theta_j)} \bigg|_{\theta = \hat{\theta}} \right).$$

Since the codelength for δ is given by $-\sum_i \ln \delta_i + O(1)$, the minimization of a type of (30) requires that

$$\sum_{t=1}^{m} L(D_t : f_{\hat{\theta}}) + \frac{m}{2} \delta^T \Sigma \delta - \frac{1}{\lambda} \sum_i \ln \delta_i$$

be minimized with respect to δ. Supposing that Σ is positive definite and that its maximum eigenvalue is uniformly upper-bounded by a constant, it can be verified that the minimum is attained by δ such that

$$\prod_{i=1}^{k} \delta_i = \Theta \left(\frac{1}{(\lambda m)^{k/2} |\Sigma|^{1/2}} \right).$$

This quantization scale δ also ensures that the minimum loss over $\Theta_k^{(m)}$ is within $O(k/\lambda)$ of that over Θ_k. This nature for the fineness of quantization may be formalized as follows:

Condition 17: For given $\lambda > 0$, there exists a quantization of $\mathcal{H}_k$ such that for some $0 < B < \infty$, for all m, for all $D^m = D_1 \cdots D_m \in \mathcal{D}^*$, the following inequality holds:

$$\min_{\theta \in \Theta_k^{(m)}} \sum_{t=1}^{m} L(D_t : f_\theta) \leq \inf_{\theta \in \Theta_k} \sum_{t=1}^{m} L(D_t : f_\theta) + \frac{Bk}{\lambda} \tag{32}$$

where $\Theta_k^{(m)}$ is a quantization of Θ_k for sample size m.

We are now ready to give a formal definition of MLC.

Algorithm MLC: Let $\mathcal{H} = \cup_k \mathcal{H}_k$ and L be given. For each k, for each m, fix a quantization $\mathcal{H}_k^{(m)}$ of $\mathcal{H}_k$ and let $\mathcal{H}^{(m)} = \cup_k \mathcal{H}_k^{(m)}$. For each m, fix $L_m : \mathcal{H}^{(m)} \rightarrow \boldsymbol{R}^+$ satisfying

$$\sum_{f \in \mathcal{H}^{(m)}} e^{-L_m(f)} \leq 1 \tag{33}$$

and λ, which may depend on m.

Input: $D^m \in \mathcal{D}^*$

Output: $\hat{f} \in \mathcal{H}^{(m)}$ such that

$$\hat{f} = \arg \min_{f \in \mathcal{H}^{(m)}} \left\{ \sum_{t=1}^{m} L(D_t : f) + \frac{1}{\lambda} L_m(f) \right\}. \tag{34}$$

In the case where the hypothesis class is a class of probability densities and the distortion measure is the logarithmic loss function, MLC coincides with the statistical model selection criterion called the *minimum description length (MDL) principle* (see [14], [16]–[19, pp. 79–92]), letting $\lambda = 1$.

MLC is closely related to Barron's *complexity regularization algorithm* [2], which we denote by CR. Barron showed that CR takes the same form as (34) with respect to the quadratic loss function and the logarithmic loss function. For other bounded loss functions, however, Barron took CR to have the following different form of

$$\hat{f} = \arg \min_{f \in \mathcal{H}^{(m)}} \left\{ \sum_{t=1}^{m} L(D_t : f) + \frac{1}{\lambda} (m L_m(f))^{1/2} \right\} \tag{35}$$

where λ is a positive constant. This form was taken to ensure bounds on the statistical risk of $\hat{f}$ by the method of analysis in [2], and no longer has interpretation as an approximation of

ESC. Unlike CR of the form of (35), MLC offers a unifying strategy that always takes the form of (34) regardless of a loss function and λ. Our analysis in the next section shows that MLC has satisfactory bounds on the statistical risk for general bounded loss functions.

C. Analysis of MLC

We analyze MLC by giving upper bounds on its statistical risk.

Theorem 3. Upper Bounds on Statistical Risk for MLC: Suppose that for some $0 < C < \infty$, for all D, for all $f \in \mathcal{H}$, $0 \le L(D : f) \le C$. Let $h(\lambda) \stackrel{\text{def}}{=} (e^{\lambda C} - 1)/C$. Assume that for the sequence $D^m = D_1, \cdots, D_m \in \mathcal{D}^*$, each D_t is independently drawn according to the unknown target distribution P. Let $\mathcal{H}^{(m)}$ be a quantization of $\mathcal{H}$. Then for any $\lambda > 0$, the statistical risk for MLC using $\mathcal{H} = \cup_k \mathcal{H}_k$ is upper-bounded as follows:

$$E[\Delta_P(\hat{f})] < \inf_{f \in \mathcal{H}^{(m)}} \left\{ C^2 h(\lambda) + \Delta_P(f) + \frac{(L_m(f)+1)}{mh(\lambda)} \right\}. \tag{36}$$

Specifically, if for given $\lambda > 0$, for each k, the quantization $\mathcal{H}_k^{(m)}$ of $\mathcal{H}_k$ satisfies Condition 17 and $L_m(f)$ takes a constant value over $\mathcal{H}_k^{(m)}$, then the statistical risk for MLC using $\mathcal{H} = \cup_k \mathcal{H}_k$ is upper-bounded as follows:

$$E[\Delta_P(\hat{f})] < \inf_{f \in \mathcal{H}} \left\{ C^2 h(\lambda) + \Delta_P(f) + \frac{(L_m(\overline{f}) + Bk + 1)}{mh(\lambda)} \right\} \tag{37}$$

where $\overline{f} \in \mathcal{H}^{(m)}$ is a truncation of $f \in \mathcal{H}$, k is the number of real-valued parameters in f, and B is a constant as in (32).

Proof: We abbreviate $\sum_{t=1}^m L(D_t : f)$ as $L(D^m : f)$. Choose $\overline{f} \in \mathcal{H}^{(m)}$ arbitrarily. Let $\hat{f}$ be an output of MLC.

First note that if $\Delta_P(\hat{f}) > \varepsilon$, then the hypothesis that attains the minimum of the quantity: $\lambda L(D^m : f) + L_m(f)$ over $\mathcal{H}^{(m)}$ lies in the range $\{f \in \mathcal{H}^{(m)} : \Delta_P(f) > \varepsilon\}$. Thus $\text{Prob}[\Delta_P(\hat{f}) > \varepsilon]$ is upper-bounded as follows:

$$\begin{aligned}
&\text{Prob}[\Delta_P(\hat{f}) > \varepsilon] \\
&\le \text{Prob}\left[\min_{f \in \mathcal{H}^{(m)}: \Delta_P(f) > \varepsilon} \{\lambda L(D^m : f) + L_m(f)\} \le \lambda L(D^m : \overline{f}) + L_m(\overline{f}) \right] \\
&= \text{Prob}\left[\max_{f \in \mathcal{H}^{(m)}: \Delta_P(f) > \varepsilon} e^{-\lambda L(D^m:f) - L_m(f)} \ge e^{-\lambda L(D^m:\overline{f}) - L_m(\overline{f})} \right] \\
&\le \sum_{f \in \mathcal{H}^{(m)}: \Delta_P(f) > \varepsilon} \text{Prob}\left[e^{-\lambda L(D^m:f) - L_m(f)} \ge e^{-\lambda L(D^m:\overline{f}) - L_m(\overline{f})} \right].
\end{aligned} \tag{38}$$

Next we evaluate the probability (38). Let E be the set of D^m satisfying the event that

$$e^{-\lambda L(D^m:f) - L_m(f)} \ge e^{-\lambda L(D^m:\overline{f}) - L_m(\overline{f})}.$$

For all $f \in \mathcal{H}^{(m)}$, we have

$$\begin{aligned}
&\text{Prob}[e^{-\lambda L(D^m:f) - L_m(f)} \ge e^{-\lambda L(D^m:\overline{f}) - L_m(\overline{f})}] \\
&= \int_{D^m \in E} dP(D^m) \\
&\le \int_{D^m \in E} dP(D^m) \frac{e^{-\lambda L(D^m:f) - L_m(f)}}{e^{-\lambda L(D^m:\overline{f}) - L_m(\overline{f})}} \\
&\le e^{-L_m(f) + L_m(\overline{f})} \int dP(D^m) e^{-\lambda L(D^m:f) + \lambda L(D^m:\overline{f})} \\
&= e^{-L_m(f) + L_m(\overline{f})} \left(\int dP(D) e^{-\lambda(L(D:f) - L(D:\overline{f}))} \right)^m.
\end{aligned} \tag{39}$$

Here we have used the independence assumption for D to derive the last equation.

We use the following key lemma to further evaluate (39).

Lemma 2: For f satisfying $\Delta_P(f) > \varepsilon$

$$\int dP(D) e^{-\lambda(L(D:f) - L(D:\overline{f}))} < \exp\left[-h(\lambda)(\varepsilon - (C^2 h(\lambda) + \Delta_P(\overline{f})))\right] \tag{40}$$

where $h(\lambda) = (e^{\lambda C} - 1)/C$.

By plugging (40) into (39), and then the resulting inequality into (38), we can upper-bound $\text{Prob}[\Delta_P(\hat{f}) > \varepsilon]$ as follows:

$$\begin{aligned}
&\text{Prob}[\Delta_P(\hat{f}) > \varepsilon] \\
&< e^{L_m(\overline{f})} e^{-mh(\lambda)(\varepsilon - (C^2 h(\lambda) + \Delta_P(\overline{f})))} \times \sum_{f \in \mathcal{H}^{(m)}: \Delta_P(f) > \varepsilon} e^{-L_m(f)} \\
&\le \exp\left[-mh(\lambda) \left(\varepsilon - \left(C^2 h(\lambda) + \Delta_P(\overline{f}) + \frac{L_m(\overline{f})}{mh(\lambda)} \right) \right) \right]
\end{aligned} \tag{41}$$

where the last inequality follows from the fact

$$\sum_{f \in \mathcal{H}^{(m)}: \Delta_P(f) > \varepsilon} e^{-L_m(f)} \le \sum_{f \in \mathcal{H}^{(m)}} e^{-L_m(f)} \le 1$$

by (33). Letting

$$\varepsilon' = \varepsilon - \left(C^2 h(\lambda) + \Delta_P(\overline{f}) + \frac{L_m(\overline{f})}{mh(\lambda)} \right)$$

(41) is written as

$$\text{Prob}\left[\Delta_P(\hat{f}) - \left(C^2 h(\lambda) + \Delta_P(\overline{f}) + \frac{L_m(\overline{f})}{mh(\lambda)} \right) > \varepsilon' \right] < e^{-mh(\lambda)\varepsilon'}.$$

Hence the statistical risk for MLC is upper-bounded as follows:

$$\begin{aligned}
E[\Delta_P(\hat{f})] &< C^2 h(\lambda) + \Delta_P(\overline{f}) + \frac{L_m(\overline{f})}{mh(\lambda)} + \int_0^\infty e^{-mh(\lambda)\varepsilon'} d\varepsilon' \\
&= C^2 h(\lambda) + \Delta_P(\overline{f}) + \frac{L_m(\overline{f}) + 1}{mh(\lambda)}.
\end{aligned} \tag{42}$$

Since (42) holds for all $\overline{f} \in \mathcal{H}^{(m)}$, we obtain (37) by minimizing the left-hand side of (42) with respect to $\overline{f}$ over $\mathcal{H}^{(m)}$. This completes the proof of (36). Bound (37) can also be proven quite similarly to (36) under Condition 17. □

Proof of Lemma 2: We start with the following two formulas.

Sublemma 1: For $0 < C < \infty$, for $\lambda > 0$

$$e^{-\lambda x} \leq 1 - \frac{1 - e^{-\lambda C}}{C} x \qquad (0 \leq x \leq C) \tag{43}$$

$$e^{-\lambda x} \leq 1 - \frac{e^{\lambda C} - 1}{C} x \qquad (-C \leq x \leq 0). \tag{44}$$

Let

$$V(D : f, \overline{f}) \stackrel{\text{def}}{=} L(D : f) - L(D : \overline{f})$$

and

$$\Delta(f \parallel \overline{f}) \stackrel{\text{def}}{=} E_P[V(D : f, \overline{f})].$$

Then $-C \leq V(D : f, \overline{f}) \leq C$. Thus making use of (43) and (44), we obtain the following upper bound on $\int dP(D) e^{-\lambda(L(D:f) - L(D:\overline{f}))}$:

$$\begin{aligned}
&\int dP(D) e^{-\lambda(L(D:f) - L(D:\overline{f}))} \\
&= \int_{D:V(D:f,\overline{f}) \geq 0} dP(D) e^{-\lambda V(D:f,\overline{f})} \\
&\quad + \int_{D:V(D:f,\overline{f}) < 0} dP(D) e^{-\lambda V(D:f,\overline{f})} \\
&\leq \int_{D:V(D:f,\overline{f}) \geq 0} dP(D) \left(1 - \frac{1 - e^{-\lambda C}}{C} V(D : f, \overline{f})\right) \\
&\quad + \int_{D:V(D:f,\overline{f}) < 0} dP(D) \left(1 - \frac{e^{\lambda C} - 1}{C} V(D : f, \overline{f})\right).
\end{aligned} \tag{45}$$

Let

$$C_1 \stackrel{\text{def}}{=} (1 - e^{-\lambda C})/C \qquad C_2 \stackrel{\text{def}}{=} (e^{\lambda C} - 1)/C$$

and

$$h(f, \overline{f}) \stackrel{\text{def}}{=} \int_{D:V(D:f,\overline{f}) \geq 0} dP(D) V(D : f, \overline{f}) (\leq C).$$

Then (45) can be further upper-bounded as follows:

$$\begin{aligned}
&\int_D dP(D) e^{-\lambda(L(D:f) - L(D:\overline{f}))} \\
&\leq 1 - C_1 \int_{D:V(D:f,\overline{f}) \geq 0} dP(D) V(D : f, \overline{f}) \\
&\quad - C_2 \int_{D:V(D:f,\overline{f}) < 0} dP(D) V(D : f, \overline{f}) \\
&= 1 - C_2 \Delta(f \parallel \overline{f}) + (C_2 - C_1) h(f, \overline{f}) \\
&\leq 1 - C_2 \Delta(f \parallel \overline{f}) + (C_2 - C_1) C \\
&= 1 - C_2 \Delta_P(f) + C_2 \Delta_P(\overline{f}) + (C_2 - C_1) C
\end{aligned} \tag{46}$$

where (46) follows from the relation: $\Delta(f \parallel \overline{f}) = \Delta_P(f) - \Delta_P(\overline{f})$. Further note that

$$(C_2 - C_1) C = (e^{\lambda C} - 1)^2 / e^{\lambda C} > 0$$

and

$$(C_2 - C_1) C = C^2 C_2^2 / e^{\lambda C} \leq C^2 C_2^2.$$

Thus we have the following inequality for any f such that $\Delta_P(f) > \varepsilon$:

$$\begin{aligned}
&\int dP(D) e^{-\lambda(L(D:f) - L(D:\overline{f}))} \\
&\qquad < 1 - C_2 \varepsilon + C^2 C_2^2 + C_2 \Delta_P(\overline{f}) \\
&\qquad \leq \exp[-C_2(\varepsilon - (C^2 C_2 + \Delta_P(\overline{f})))]
\end{aligned} \tag{47}$$

where (47) follows from the fact that for any $A > 0, 1 - Ax \leq e^{-Ax}$. Rewriting C_2 in (47) as $h(\lambda)$ yields (40). This completes the proof of Lemma 2. □

Note that bound (37) is general in the sense that it holds for all $\lambda > 0$, while Barron's CR of the form of (35) has an upper bound on its statistical risk

$$\inf_{f \in \mathcal{H}^{(m)}} \left\{ \Delta_P(f) + \text{const} \left(\frac{L_m(f)}{m} \right)^{1/2} \right\} \tag{48}$$

under some constraints of λ. As will be seen in Corollary 1, however, MLC also leads to the square-root regularization term (with respect to m) after making necessary adjustments to λ to obtain the least upper bound on its statistical risk. In the end, MLC has the same performance as CR, while MLC has generality and allowance of the criterion to take the form of (34) rather than (35). (Note: A. R. Barron recently informed the author that the square root in the regularization term in (35) was not needed for the statistical risk bounds similar to those given in [2], as pointed out to him by Bartlett and Lee.)

Let $\mathcal{F}$ be a set of all functions from $\mathcal{X}$ to $\mathcal{Z}$ or a set of conditional probability densities or conditional probability mass functions over $\mathcal{Y}$ for given $X \in \mathcal{X}$. Assume that for a given target distribution P, there exists a function f^* that attains the minimum of $E_P[L(D : f)]$ over all f in $\mathcal{F}$. Letting $\mathcal{H}_k \subset \mathcal{F}$ be a k-dimensional parametric class, the *parametric case* is the case where f^* is in $\mathcal{H}_k$ for some finite k. The *nonparametric case* is the case where f^* is not in $\mathcal{H}_k$ for any $k < \infty$. Below, as a corollary of Theorem 3, we give upper bounds on the statistical risk for MLC both for the parametric and nonparametric cases.

Corollary 1: Suppose that for each k, $\mathcal{H}_k$ and L satisfy Condition 17 and that for the quantization of $\mathcal{H}_k$ satisfying (32), any quantization scale $\delta = (\delta_1, \cdots, \delta_k)$ satisfies $\prod_{i=1}^k \delta_i = \Theta(1/(\lambda m)^{k/2})$. Suppose also that $L_m(f)$ takes a constant value over $\mathcal{H}_k^{(m)}$.

Parametric case: Assume that for the target distribution P, for some $k^* < \infty$, f^* is in $\mathcal{H}_{k^*}$ and is written as f_{θ^*}. Then letting

$$\lambda = \frac{1}{C} \ln \left(1 + C \left(\frac{\ln m}{m} \right)^{1/2} \right) = O\left(\left(\frac{\ln m}{m} \right)^{1/2} \right)$$

we have the following upper bound on the statistical risk for MLC:

$$E[\Delta_P(\hat{f})] = O\left(\left(\frac{\ln m}{m} \right)^{1/2} \right). \tag{49}$$

Nonparametric case: Assume that for the target distribution P, for some $\alpha > 0$, for each k, the optimal hypothesis f^* can be approximated by a k-dimensional subclass $\mathcal{H}_k$ of $\mathcal{H}$ with error $\inf_{f \in \mathcal{H}_k} \Delta_P(f) = O(1/k^\alpha)$. Then letting

$$\lambda = \frac{1}{C} \ln \left(1 + C \left(\frac{\ln m}{m} \right)^{1/2} \right)$$

we have the following upper bound on the statistical risk for MLC:

$$E[\Delta_P(\hat{f})] = O\left(\left(\frac{\ln m}{m} \right)^{\alpha/(2(\alpha+1))} \right). \tag{50}$$

In the special case, where α is known to MLC in advance, letting $\lambda = (1/C) \ln(1 + C((\ln m)/m)^{\alpha/(2\alpha+1)})$, we have the following upper bound on the statistical risk for MLC:

$$E[\Delta_P(\hat{f})] = O\left(\left(\frac{\ln m}{m} \right)^{\alpha/(2\alpha+1)} \right). \tag{51}$$

Proof: First consider the parametric case where f^* is written as $f_{\theta^*} \in \mathcal{H}_{k^*}$ for some k^*. For each k, let $\mathcal{H}_k^{(m)}$ be a quantization of $\mathcal{H}_k$ with quantization scale δ such that $\prod_{i=1}^k \delta_i = \Theta(1/(\lambda m)^{k/2})$ for sample size m. Let $\overline{f}^*$ be the truncation of $f^* \in \mathcal{H}_{k^*}$. Then we see that

$$L_m(\overline{f}^*) = O(k^* \ln(m\lambda)) = O(k^* \ln m).$$

If we set $\lambda = (1/C) \ln(1 + C((\ln m)/m)^{1/2})$, then we have $h(\lambda) = O(((\ln m)/m)^{1/2})$. Under Condition 17 we can use (37) and the fact that $\Delta_P(f^*) = 0$ to upper-bound the statistical risk as follows:

$$\begin{aligned} E[\Delta_P(\hat{f})] &< C^2 h(\lambda) + \Delta_P(f^*) + \frac{(L_m(\overline{f}^*) + Bk^* + 1)}{mh(\lambda)} \\ &= O\left(\left(\frac{\ln m}{m} \right)^{1/2} \right) \end{aligned}$$

which yields (49).

Next consider the nonparametric case where

$$\inf_{f \in \mathcal{H}_k} \Delta_P(f) = O(1/k^\alpha).$$

We can use (37) to obtain the following bound on the statistical risk for MLC:

$$\begin{aligned} E[\Delta_P(\hat{f})] &< \min_k \inf_{f_\theta \in \mathcal{H}_k^{(m)}} \left\{ C^2 h(\lambda) + \Delta_P(f_\theta) + \frac{(L_m(\overline{f}_\theta) + Bk + 1)}{mh(\lambda)} \right\} \\ &= O\left(\min_k \left(\left(\frac{\ln m}{m} \right)^{1/2} + \frac{1}{k^\alpha} + k \left(\frac{\ln m}{m} \right)^{1/2} \right) \right) \\ &= O\left(\left(\frac{\ln m}{m} \right)^{\alpha/2(\alpha+1)} \right) \end{aligned} \tag{52}$$

which yields (50). The minimum in (52) is attained by $k = O((m/\ln m)^{1/2(\alpha+1)})$. Bound (51) can be obtained similarly with (50). □

For the parametric case, the convergence rate bound (49) coincides with that obtained for Barron's CR with respect to m (see [2]). This bound is fastest to date. For the nonparametric case, (51) is slightly better than (50) and coincides with that obtained for CR in [2], which is fastest to date. Note that (51) is attained by CR even if α is not known to CR in advance.

For the quadratic loss function and logarithmic loss function, MLC takes exactly the same form as CR. Then (49) and (50) can be improved to $O((\ln m)/m)$ for the parametric case and $O(((\ln m)/m)^{\alpha/(\alpha+1)})$ for the nonparametric case, respectively, where α is an index as in Corollary 1 (see the analysis in [2]).

V. Concluding Remarks

We have introduced ESC as an extension of stochastic complexity to the decision-theoretic setting where a general real-valued function is used as a hypothesis and a general loss function is used as a distortion measure. Through ESC we have given a unifying view of the design and analysis of the learning algorithms which have turned out to be most effective in batch-learning and on-line prediction scenarios.

For the on-line prediction scenario, a sequential realization of ESC induces the multiplicatively weight-decaying algorithm called the aggregating strategy AGG. This corresponds to the fact that a sequential realization of SC induces the Bayes algorithm for the specific case where the hypothesis class is a class of probability densities and the distortion measure is the logarithmic loss. We have derived upper bounds on both the expected and worst case cumulative losses for AGG, for the case where the hypothesis class is specified by a continuous parameter. These bounds are the first ones derived in a general decision-theoretic setting. Haussler *et al.* [11] derived a worst case lower bound on the cumulative loss for any on-line prediction algorithm using a finite hypothesis class. They showed that a version of ESC defined relative to a finite hypothesis matches their lower bound within error $o(1)$. However, for the case where the hypothesis class is continuous, it is an open problem to derive a tight lower bound on the cumulative loss. Recently this problem has been addressed in [29] and it has turned out that under certain conditions for a hypothesis class, there exists a lower bound on the worst case cumulative loss for any on-line prediction algorithm using the hypothesis class, which matches ESC within $o(\ln m)$ for sample size m.

For the batch-learning scenario, a batch-approximation of ESC using a single hypothesis induces the learning algorithm MLC, which is a formal extension of the MDL learning algorithm. We have derived upper bounds on the statistical risk for MLC with respect to general bounded loss functions. It has turned out that MLC has the least upper bounds (to date) on the statistical risk by tuning λ optimally. It remains for future study to derive a tight lower bound on the statistical risk for MLC to compare it with our upper bounds.

Throughout this paper it has turned out that λ in the definition of ESC plays an important role both for on-line prediction and batch learning. In the on-line learning scenario, λ is determined so that AGG never fails, while in the batch-learning scenario, it is determined so that the rate of the convergence for MLC is made fastest. Hence, λ must be tuned

depending on the learning scenario although there currently seems to be a lack of unifying understanding of λ.

ESC might often be computationally or analytically hard to compute when the hypothesis class is specified by a complicated constraint for real-valued parameters, e.g., neural networks and hierarchically parametric models. There remains an important computational issue of how to approximate ESC, considering the tradeoff between the accuracy of the approximation to ESC and its computational efficiency.

ACKNOWLEDGMENT

The author wishes to thank Andrew Barron and Vijay Balasubramanian for very helpful comments and suggestions. He also wishes to thank Peter Grunwald and Jason Catlett for reading an earlier draft of this paper as well as Jun-ichi Takeuchi and three anonymous reviewers for their valuable comments.

REFERENCES

[1] V. Balasubramanian, "Statistical inference Occam's razor, and statistical mechanics on the space of probability distributions," *Neural Comput.*, vol. 9, pp. 349–368, Feb. 1997.

[2] A. R. Barron, "Complexity regularization with application to artificial neural networks," in *Nonparametric Functional Estimation and Related Topics*, G. Roussas, Ed. Norwell, MA: Kluwer, 1991, pp. 561–576.

[3] A. R. Barron and T. Cover, "Minimum complexity density estimation," *IEEE Trans. Inform. Theory*, vol. 37, pp. 1034–1054, July 1991.

[4] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis.* New York: Springer-Verlag, 1985.

[5] N. G. De Bruijn, *Asymptotic Methods in Analysis.* New York: Dover, 1958.

[6] B. S. Clarke and A. R. Barron, "Information-theoretic asymptotics of Bayes methods," *IEEE Trans. Inform. Theory*, vol. 36, pp. 453–471, May 1990.

[7] N. Cesa-Bianchi, Y. Freund, D. P. Helmbold, D. Haussler, R. Schapire, and M. K. Warmuth, "How to use expert advice," in *Proc. 25th ACM Symp. Theory of Computing.* ACM Press, 1993, pp. 429–438.

[8] A. Dawid, "Statistical theory: The prequential approach," *J. Roy. Statist. Soc. A*, pp. 278–292, 1991.

[9] Y. Freund, "Predicting a binary sequence almost as well as the optimal biased coin," in *Proc. 9th ACM Conf. Computational Learning Theory.* ACM Press, 1996, pp. 89–98.

[10] D. Haussler, "Generalizing the PAC model for neural net and other learning applications," *Inform. Comput.*, vol. 100, pp. 78–150, Sept. 1992.

[11] D. Haussler, J. Kivinen, and M. Warmuth, "Tight worst-case loss bounds for predicting with expert advice," in *Computational Learning Theory: 2nd European Conf., EuroCOLT'95.* New York: Springer-Verlag, 1995, pp. 69–83.

[12] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," in *Proc. 30th IEEE Symp. the Foundations of Computer Science*, 1989, pp. 256–261.

[13] J. Kivinen and M. Warmuth, "Using experts for predicting continuous outcomes," in *Computational Learning Theory: EuroCOLT'93.* Oxford, U.K.: Oxford Univ. Press, 1994, pp. 109–120.

[14] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.

[15] ——, "A universal data compression system," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 656–664, Sept. 1983.

[16] ——, "Universal coding, information, prediction, and estimation," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 629–636, July 1984.

[17] ——, "Stochastic complexity and modeling," *Ann. Statist.*, vol. 14, pp. 1080–1100, Sept. 1986.

[18] ——, "Stochastic complexity," *J. Roy. Statist. Soc. B*, vol. 49, no. 3, pp. 223–239, 1987.

[19] ——, *Stochastic Complexity in Statistical Inquiry.* Singapore: World Scientific, 1989.

[20] ——, "Fisher information and stochastic complexity," *IEEE Inform. Theory*, vol. 42, pp. 40–47, Jan 1996.

[21] C. E. Shannon, "A mathematical theory of communications," in *Bell Syst. Tech. J.*, vol. 47, pp. 147–157, 1948.

[22] V. G. Vovk, "Aggregating strategies," in *Proc. 3rd Annu. Work. Computational Learning Theory.* Morgan Kaufmann, 1990, pp. 371–386.

[23] K. Yamanishi, "A learning criterion for stochastic rules," *Machine Learning*, vol. 9, pp. 165–203, 1992.

[24] ——, "Generalized stochastic complexity and its applications to learning," in *Proc. 1994 Conf. Information Science and Systems*, 1994, vol. 2, pp. 763–768.

[25] ——, "The minimum L-complexity algorithm and its applications to learning nonparametric rules," in *Proc. 7th Annu. ACM Conf. Computational Learning Theory.* ACM Press, 1994, pp. 173–182.

[26] ——, "Probably almost discriminative learning," *Machine Learning*, vol. 18, pp. 23–50, 1995.

[27] ——, "A loss bound model for on-line stochastic prediction algorithms," *Inform. Comput.*, vol. 119, no. 1, pp. 39–54, May 1995.

[28] ——, "On-line maximum likelihood prediction with respect to general loss functions," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 105–118, 1997.

[29] ——, "Minimax relative loss analysis for sequential prediction algorithm using parametric hypotheses," in *Proc. 11th Annu. Conf. Computational Learning Theory*, 1998, to be published.